De Montfort University

Faculty of Computing, Engineering and Media

Machine Learning for Human Activity Detection in Smart Homes

Author: Anastasios Vafeiadis

Affiliation: Center for Research & Technology Hellas - Information Technologies Institute

Sponsorship: European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No. 676157, project ACROSSING

First Supervisor:
Professor Liming Chen
Second Supervisor:
Professor Raouf Hamzaoui

External Advisers:
Dr. Konstantinos Votis
Dr. Dimitrios Giakoumis
Dr. Dimitrios Tzovaras

A thesis submitted for the degree of

PhD in Computer Science

August 28, 2020

Dedicated to my parents and my grandmother, Maria.

Abstract

Recognizing human activities in domestic environments from audio and active power consumption sensors is a challenging task since on the one hand, environmental sound signals are multi-source, heterogeneous, and varying in time and on the other hand, the active power consumption varies significantly for similar type electrical appliances.

Many systems have been proposed to process environmental sound signals for event detection in ambient assisted living applications. Typically, these systems use feature extraction, selection, and classification. However, despite major advances, several important questions remain unanswered, especially in real-world settings. A part of this thesis contributes to the body of knowledge in the field by addressing the following problems for ambient sounds recorded in various real-world kitchen environments: 1) which features, and which classifiers are most suitable in the presence of background noise? 2) what is the effect of signal duration on recognition accuracy? 3) how do the SNR and the distance between the microphone and the audio source affect the recognition accuracy in an environment in which the system was not trained? We show that for systems that use traditional classifiers, it is beneficial to combine gammatone frequency cepstral coefficients and discrete wavelet transform coefficients and to use a gradient boosting classifier. For systems based on deep learning, we consider 1D and 2D CNN using mel-spectrogram energies and mel-spectrograms images, as inputs, respectively and show that the 2D CNN outperforms the 1D CNN. We obtained competitive classification results for two such systems and validated the performance of our algorithms on public datasets (Google Brain/TensorFlow Speech Recognition Challenge and the 2017 Detection and Classification of Acoustic Scenes and Events Challenge).

Regarding the problem of the energy-based human activity recognition in a household environment, machine learning techniques to infer the state of household appliances from their energy consumption data are applied and rule-based scenarios that exploit these states to detect human activity are used. Since most activities within a house are related with the operation of an electrical appliance, this unimodal approach has a significant advantage using inexpensive smart plugs and smart meters for each appliance. This part of the thesis proposes the use of unobtrusive and easy-install tools (smart plugs) for data collection and a decision engine that combines energy signal classification using dominant classifiers (compared in advanced with grid search) and a probabilistic measure for appliance

usage. It helps preserving the privacy of the resident, since all the activities are stored in a local database.

DNNs received great research interest in the field of computer vision. In this thesis we adapted different architectures for the problem of human activity recognition. We analyze the quality of the extracted features, and more specifically how model architectures and parameters affect the ability of the automatically extracted features from DNNs to separate activity classes in the final feature space. Additionally, the architectures that we applied for our main problem were also applied to text classification in which we consider the input text as an image and apply 2D CNNs to learn the local and global semantics of the sentences from the variations of the visual patterns of words. This work helps as a first step of creating a dialogue agent that would not require any natural language preprocessing.

Finally, since in many domestic environments human speech is present with other environmental sounds, we developed a Convolutional Recurrent Neural Network, to separate the sound sources and applied novel post-processing filters, in order to have an end-to-end noise robust system. Our algorithm ranked first in the Apollo-11 Fearless Steps Challenge.

Declaration

I declare that the material presented in this thesis consists of original work undertaken solely by myself. Information derived from the published and unpublished work of others has been properly referenced. The material has not been submitted in substantially the same form for the award of a higher degree elsewhere.

Anastasios Vafeiadis

August 28, 2020

Acknowledgements

First of all, I am profoundly thankful to my local advisers Dr. Konstantinos Votis, Dr. Dimitrios Giakoumis and Dr. Dimitrios Tzovaras for selecting me amongst other candidates as a recipient of the Marie Skłodowska-Curie fellowship. I am grateful to have Professor Liming Chen and Professor Raouf Hamzaoui as supervisors in this PhD. Their knowledge in the field and the constant support helped me mature as a researcher and as a person.

Secondly, I would like to thank all my fellow Early Stage Researchers whom I am privileged to call them friends and not just colleagues. We had a great collaboration and exchange of knowledge throughout the project.

I would also like to thank my friend Eleftherios Fanioudakis whom I met when participating at the Google Brain TensorFlow speech recognition challenge. At the speech activity detection task of the Fearless Steps Challenge, we achieved the first place among 27 submissions, after endless Skype chats. This collaboration has shown to me that when people are fully dedicated into solving a problem, they can accomplish great things, even if they have never met in person.

Fourth, I would like to thank all my friends and colleagues at CERTH and especially Aris and Thanasis that provided constant moral support during this PhD and helped me look at life from another perspective. I consider myself lucky that I have met these people.

Lastly, I would like to thank my parents, Petros and Evangelia and my sister Eleni. I would not be able to reach this point in my life without them. They were always next to me and they provided the utmost support at every step, especially of my academic career.

Contents

Abstract **2**

Declaration **4**

Acknowledgements **5**

Contents **6**

List of Figures **10**

List of Tables **12**

Acronyms **13**

1 Introduction **16**
1.1 Project Background 16
1.2 Research Problems and Questions 17
1.3 Project Objectives 20
1.4 Proposed Approaches 21
1.5 Contributions and Paper Publications 22
1.6 Outline of the Thesis 25

2 Background **27**
2.1 Introduction to Human Activity Recognition 27
2.2 Audio-based Event Detection and Activity Recognition 28
2.2.1 Computational Auditory Scene Analysis 29
2.2.2 Acoustic Scene Recognition 30
2.2.3 Deep Learning in Audio-based Event Detection 32
2.2.4 Privacy Issues 33
2.3 Energy-based Event Detection and Activity Recognition 33
2.3.1 Statistical and Traditional Machine Learning Approaches in Energy-based Event Detection 33
2.3.2 Deep Learning in Energy-based Event Detection 35

2.3.3 Privacy Issues . . . 35
2.4 Comparison of Human Crafted Engineering Features and Learnt Features of a CNN for Activity Recognition . . . 36
2.4.1 Inertial sensor . . . 38
2.4.2 Audio Features . . . 38

3 Audio-based Event Detection 40
3.1 Introduction . . . 40
3.2 Acoustic Event Detection Frameworks . . . 42
3.2.1 Signal Acquisition . . . 42
3.2.2 Data Augmentation . . . 45
3.2.3 Feature Extraction . . . 46
3.2.3.1 Mel-Frequency Cepstral Coefficients . . . 46
3.2.3.2 Discrete Wavelet Transform . . . 47
3.2.3.3 Zero-Crossing Rate . . . 47
3.2.3.4 Spectral Roll-off . . . 47
3.2.3.5 Spectral Centroid . . . 48
3.2.3.6 Gammatone Frequency Cepstral Coefficients . . . 49
3.2.4 Feature Selection . . . 50
3.2.4.1 Feature Aggregation . . . 50
3.2.4.2 Sequential Backward Selection . . . 50
3.2.4.3 Principal Component Analysis . . . 51
3.2.5 Activity Classification . . . 51
3.3 Results . . . 52
3.3.1 Comparison of the recognition accuracy of traditional classifiers against CNNs for AED . . . 53
3.3.2 Fusion of features for the first AED approach . . . 56
3.3.3 Recognition accuracy as a function of the audio sample duration . . . 57
3.3.4 Dependence of recognition accuracy on certain distance and SNR in a new environment . . . 58
3.3.5 Tests with activity that was not included in the training set (coffee machine) using the second AED system . . . 60
3.4 Conclusions . . . 60

4 Convolutional Recurrent Neural Networks for Speech Activity Detection 62
4.1 Introduction . . . 62
4.2 Proposed Approach for Speech Activity Detection . . . 64
4.2.1 Feature Extraction . . . 64
4.2.2 Convolutional Recurrent Neural Networks Description . . . 64
4.3 Evaluation and Analysis . . . 66
4.3.1 Network Architectures . . . 66
4.3.2 Network Training . . . 67
4.4 Results . . . 68

4.5 Conclusions . . . 71

5 Energy-based Event Detection 72
5.1 Introduction . . . 72
5.2 Data Collection and Analysis . . . 73
5.2.1 Data Collection Infrastructure . . . 73
5.2.2 Data Pre-Processing . . . 75
5.2.3 Appliance State Proportion Feature . . . 77
5.3 Decision Engine for Human Activity Recognition . . . 78
5.4 Results . . . 80
5.5 Conclusions . . . 81

6 Investigation of 2D Convolutional Neural Networks 82
6.1 Introduction . . . 82
6.2 Comparison of Human Crafted Engineering Features and Learnt Features of a CNN . . . 83
6.2.1 Automatic Feature Extraction . . . 84
6.2.2 Approach and Experiment . . . 84
6.2.3 Evaluation . . . 87
6.2.3.1 Extracted Features . . . 88
6.2.3.2 Experimental Environment . . . 88
6.2.4 Results . . . 88
6.2.4.1 IMU CNN Features . . . 88
6.2.4.2 Audio CNN Features . . . 91
6.2.5 Discussion . . . 93
6.2.5.1 IMU CNN Features . . . 93
6.2.5.2 Audio CNN Features . . . 96
6.2.6 Conclusions . . . 97
6.3 2D Convolutional Neural Networks for dialogue modelling . . . 98
6.3.1 Related Work . . . 100
6.3.2 Proposed Method . . . 101
6.3.2.1 Models . . . 102
6.3.2.2 Data Augmentation . . . 103
6.3.3 Results . . . 104
6.3.3.1 Text classification . . . 104
6.3.3.2 Dialogue modeling . . . 106
6.3.4 Conclusions . . . 107

7 Conclusions 109
7.1 Summary of Contributions . . . 109
7.2 Limitations of this Work . . . 110
7.3 Open Issues and Future Work . . . 112
7.4 Concluding Remarks . . . 113

A Ethics Approval AKTIOS 115

Bibliography **139**

List of Figures

2.1 CASA System Overview 30
2.2 Intersection of Audio Event Recognition with the wider scientific field 31

3.1 First proposed AED approach. 42
3.2 Experimental Setup 43
3.3 MFCC Feature Extraction [127] 46
3.4 SR comparison between the sound of the kitchen sink (top) and the sound of a violin (bottom). The x-axis shows the time in s 48
3.5 SC comparison between the sound of the kitchen sink (top) and the sound of a violin (bottom). The x-axis shows the time in s 49
3.6 Second proposed AED approach 51
3.7 ROC curves for the selected classifiers. Classes 0, 1, 2, 3, 4, 5 and 6 correspond to boiling, cutting bread, dishwasher, doing the dishes, frying, operating the kitchen faucet and mixer, respectively 55
3.8 Recognition accuracy for different audio features using Gradient Boosting. The results are after the aforementioned feature selection. 56
3.9 Recognition accuracy (using the Gradient Boosting classifier and the CNN) as a function of the sample duration 58

4.1 2D CRNN architecture with STFT spectrogram magnitude representation as input. The arrows around the Conv2D blocks indicate the same max-pooling operation applied after each convolutional layer 65
4.2 1D CRNN architecture with raw waveform as input. The arrows around the Conv1D blocks indicate the same max-pooling operation applied after each convolutional layer 67
4.3 Examples of speech and non-speech activity detection of 1D and 2D (STFT) CRNN architectures with the ground truth of the development dataset 69
4.4 SAD results for the 5-folds and the ensembled majority on an example of the evaluation dataset (no ground truth given) using 2D (STFT) CRNN 70
4.5 SAD results using the moving average filter of the convolutions (temporal smoothing) after averaging the 5 folds 70

5.1 Data collection environments 74
5.2 Data collection infrastructure 75

5.3 Power consumption plots of the selected appliances. Top-left is the cooker hood, top-right is the oven, bottom-left is the fridge and bottom-right the dishwasher. 76
5.4 Fridge power consumption from House B 77
5.5 Overview of energy sensor-based decision engine human activity detection 79

6.1 A typical CNN architecture taking IMU raw data as input will include multiple convolutional layers, with each layer followed by a max-pooling operation. The output of last convolutional layer is then flattened. The vector obtained corresponds to a feature vector automatically extracted. Finally, softmax is generally applied to connect to the output layer in multi-class problems. 86
6.2 Cross-validation of a CNN automatic feature extractor: (top) two CNN feature extractors are trained using datasets collected in controlled conditions (UCI-HAR and the DCASE 2017 development datasets), (middle) the CNN model is used as feature extractor, and training data from the final real-world dataset is used to train IMU-HAR and AUDIO-HAR models, (bottom) finally results are evaluated over the final test data. 87
6.3 Architecture used to compare HCF and CNN features: taking features in input, one dense layer (64 nodes) and 6 classes (for the inertial sensors) and 3 classes (for the audio sensors) on the output layer. 89
6.4 Visual comparison of 1D CNN architectures using $n = 1, 2$ and 4 layers with a kernel $k = 2$ (on the left), and kernel size $k = 2, 16, 32$ using $n = 3$ layers (on the right). 90
6.5 Visual comparison of (a) HCF, and (b) CNN using 3 layers. 91
6.6 Normalized confusion matrices obtained using (a) HCF and (b) using CNN. 92
6.7 Visual comparison (first with second PCA and first with third PCA) of 2D CNN architectures using $n = 1$ (top left), $n = 2$ (mid left) and $n = 4$ (bottom left) layers with a kernel $k = 2$, and kernel size $k = 8$ (top right), $k = 16$ (mid right), $k = 32$ (bottom right) with $n = 2$ layers for the audio dataset. 94
6.8 Visual comparison of HCF (top) and CNN using 2 layers (bottom) for the audio sensor. 95
6.9 Un-normalized confusion matrices obtained using (a) HCF and (b) using CNN for the audio dataset. 96
6.10 Top: Sogou News dataset with Chinese characters. Bottom: Sogou News dataset with pinyin (romanization of the Chinese characters based on their pronunciation) 102
6.11 Proposed model: 3 convolutional layers consisting of 32 filters with a kernel of size 5×5 each, are followed by 4 convolutional layers consisting of 64 filters with a kernel of size 5×5 each. A linear fully connected layer and a classification output layer complete the model. 103

List of Tables

3.1 Number of recordings of each class from different sources 44
3.2 Classifier Performance Comparison 53
3.3 McNemar's test results 54
3.4 McNemar's test on the features 57
3.5 Confusion matrix using Gradient Boosting for the classes of the framework in a new environment (not included in the training dataset). The distance between the microphone and each activity was 3 m . . 58
3.6 Confusion matrix using CNN for the classes of the framework in a new environment (not included in the training dataset). The distance between the microphone and each activity was 3 m 59
3.7 Recognition accuracy of Gradient Boosting and CNN according to distances and SNRs 59

4.1 Performance of different architectures using DCF as a metric on the development dataset. No temporal collars are used 68

5.1 Accuracy of proposed decision engine 79

6.1 Precision, Recall and F-Score obtained on UCI-HAR Dataset using HCF and CNN features obtained with different parameters using accelerometer only. 92
6.2 Precision, Recall and F-Score obtained on UCI-HAR Dataset using HCF and CNN features obtained with different parameters using accelerometer and gyroscope. 92
6.3 Precision, Recall and F-Score (averaged over 4-folds) obtained on the DCASE 2017 development using different parameters. 95
6.4 Results of Latin and Chinese text classification in terms of held-out accuracy. Worst-Best Performance reports the results of the worst and best performing baselines from Table 4 of Zhang et al. [208] and Conneau et al. [209]. Results reported for *TI-CNN* were obtained in 10 epochs 104
6.5 Generated text for testing. The following text samples were not seen during the training 105
6.6 Facebook bAbI Dialogue Task 4 107

Acronyms

AAL Ambient Assisted Living. 17, 18, 36, 72

ADL Activities of Daily Living. 37

AED Acoustic Event Detection. 40, 42, 45, 46, 50–54, 56, 58, 60

ANN Artificial Neural Network. 28, 79

ARC Activity Recognition Chain. 37

ASR Acoustic Scene Recognition. 18, 19, 30, 31, 62

BPN Back-Propagation Network. 79, 80

CASA Computational Auditory Scene Analysis. 29, 30

CNN Convolutional Neural Network. 2, 3, 22, 23, 25, 26, 29, 32, 35–39, 41, 42, 51, 52, 54, 56, 58–60, 63, 65, 66, 82–86, 88, 89, 91, 93, 96–104, 106–114

CRF Conditional Random Field. 33

CRNN Convolutional Recurrent Neural Network. 23, 26, 63–68, 70, 71, 110, 113

DBN Deep Belief Network. 28

DCF Decision Cost Function. 63, 65, 67, 70

DCT Discrete Cosine Transform. 50

DFT Discrete Fourier Transform. 38, 49

DL Deep Learning. 36–39, 63, 84

DNN Deep Neural Networks. 3, 17, 18, 29, 32

DSP Digital Signal Processing. 44

DT Decision Trees. 79, 80

DWT Discrete Wavelet Transform. 42, 47, 50, 56, 57, 60

ELU Exponential Linear Unit. 51, 52

ERB Equivalent Rectangular Bandwidth. 31, 56

FFT Fast Fourier Transform. 54, 64, 66, 85, 91, 93

GB Gradient Boosting. 79, 80

GFCC Gammatone Frequency Cepstral Coefficient. 31, 42, 50, 56, 60

GMM Gaussian Mixture Model. 63

GPS Global Positioning System. 38, 87

GPU Graphic Processing Unit. 32, 104

GRU Gated Recurrent Unit. 64–66, 98, 101

HAR Human Activity Recognition. 25, 28, 36–38, 40

HCF Human Crafted Features. 23, 36–38, 83–86, 88, 89, 91, 96, 97, 110

HMM Hidden Markov Model. 28, 29, 32–34

ICT Information and Communications Technology. 28

IMU Inertial Measuring Unit. 25, 83, 85, 87–89, 113

IoT Internet of Things. 21, 35

kNN k-Nearest Neighbors. 28, 51

LR Logistic Regression. 79, 80

LSTM Long Short-Term Memory. 36, 66, 67, 98

MCC Matthews Correlation Coefficient. 79, 80

MEMS Micro Electrical-Mechanical System. 17, 44, 60

MFCC Mel-frequency Cepstral Coefficient. 10, 28, 30, 32, 39, 42, 46, 47, 50, 56, 57, 66, 83–85, 97

ML Machine Learning. 36

MP Matching Pursuit. 30, 31

NB Naive Bayes. 79

NILM Non-Intrusive Load Monitoring. 34, 35, 72

NLP Natural Language Processing. 23, 26, 82, 98, 100, 101, 105–108, 114

NUC Next Utterance Classification. 101, 107

nZEB nearly Zero Energy Building. 43

OCR Optical Character Recognition. 23, 98

OOV Out-Of-Vocabulary. 107

PCA Principal Component Analysis. 51, 90, 93

RBF Radial Basis Function. 51, 53, 79, 80

RBS Rule-Based Scenarios. 78

ReLU Rectified Linear Unit. 52, 66, 93

RF Random Forest. 79, 80

RNN Recurrent Neural Networks. 63–66, 82, 98, 99, 114

ROC Receiver Operating Characteristic. 54

SAD Speech Activity Detection. 62, 63, 65, 68, 70, 71, 111, 113

SBS Sequential Backward Selection. 50, 51

SC Spectral Centroid. 10, 42, 48–50

SNR Signal-to-noise Ratio. 2, 19, 23, 29, 40, 41, 44, 52, 59, 62

SR Spectral Roll-Off. 10, 42, 47, 48, 50

STFT Short-time Fourier Transform. 23, 64, 66, 68, 70, 71, 110

SVM Support Vector Machine. 27, 29, 34, 51, 53, 79, 80

VAD Voice Activity Detection. 66, 68, 71

ZCR Zero-Crossing Rate. 42, 47, 50

Chapter 1

Introduction

1.1 Project Background

Assisted living in smart homes can change the way millions of elderly people live, manage their conditions and maintain well-being [1]. This could support the ageing population to live longer independently and to enjoy comfort and quality of life in its private environments. While current monitoring and assistive technologies are selectively deployed due to high cost, limited functionality and interoperability issues, future smart homes could leverage cheap ubiquitous sensors, interconnected smart objects, packaged with robust context interference and interaction techniques [2]. The next generation of smart home technologies will be adaptive to fit versatile living environments, and interoperable for heterogeneous applications. In addition, a service-oriented cloud-based system architecture will support reconfiguration and modular design that is essential to empower care providers to customize solutions.

With the increasing ageing population and the growing demand on novel health care models, research on smart homes for independent living, self-management and well-being has intensified over the last decade due to the wide availability of affordable sensing and effective processing technologies. Yet, it remains a challenge to develop and deploy smart home solutions that can handle everyday life situations and support a wide range of users and care applications. Smart home technologies must be interoperable for seamless technology integration and rapid application development, and adaptable for easy deployment and management, achieved by thorough testing and validation in multiple application scenarios. This requires a joint multidisciplinary cross-sector effort of research and development.

A monitoring system within a smart home checks the daily activities and decides whether the behavior of the residents is regular or irregular. The success of such a system depends on understanding the normal lifestyle and the degree to which the behavior of the elderly has deviated from what is defined as normal.

This project developed an in-depth understanding of automatic activity detection and context inference within smart home environments of healthy people, but also patients with mild cognitive impairments, dementia and Parkinson's disease. Specifically, the project investigated the use of wireless acoustic and energy meter sensor network for daily activity monitoring and detection. It developed: i) a holistic framework that integrates low-cost sensors (power consumption smart meters and MEMS microphones) for activity detection and behavior modeling of elderly people suffering with dementia or Parkinson, ii) algorithms for audio-based event classification using statistical feature analysis, data augmentation, feature extraction techniques and deep learning, iii) an automatic activity detection framework to recognize events and create user profile, with daily activities, based on acoustic and active power consumption sensors, iv) associated technologies and an integration of the developed algorithms on a low-power single board computer for evaluation by real users from care homes and living lab environments.

1.2 Research Problems and Questions

This research seeks to investigate four main aspects of building an AAL system. These aspects include building a holistic framework using open source and off-the-shelf products (e.g., MEMS microphones, Raspberry Pi), speech/non-speech activity detection, energy-based event detection and adaptation of algorithms in a different domain (e.g., CRNNs used for object detection can also be used for speech activity detection).

A few research questions regarding the problem of audio-based event detection are related to the distance of the sensor from the target appliance related to an activity, the selection of features and classifiers for an indoor environment and the parameters of DNN and their impact on recognition accuracy. Regarding speech/non-speech activity detection, the fundamental problem is the design of a general system that would perform equally well in a very noisy and a quiet environment. For the problem of energy-based event detection, a computationally inexpensive system is needed, that would make use of the active power consumption and discard other features such as statistical time-series features (skewness,

kurtosis). Finally, there is the research problem of adapting a DNN architecture and training it from scratch for a different task (e.g., natural language processing) and pre-training on a dataset of a particular domain and testing on a different dataset, but in the same domain. In this way, one can examine how well a DNN system generalizes and which parameters would require fine tuning.

Knowing the activity of occupants in a building at any given time is fundamental for the effective management of various building operation functions ranging from energy savings to security targets, especially in complex buildings with different internal kind of use (e.g., an office building, a hotel). As the activities of occupants within the building vary throughout the day, it is difficult to characterize the different activities in different time periods. In general, activity monitoring in buildings is of high interest, since it significantly contributes to the improvement of a building's energy efficiency [3] and increases the quality of life of people in AAL environments [4]. Therefore, there is a need for detailed activity knowledge.

Human activity can be estimated using various sources, such as movement sensors [5], occupancy sensors [6], cameras [7], audio [8], as well as appliance current consumption [9].

Let us imagine that one is listening to an audio recording, in which one can hear boiling sounds, frying sounds, sounds of cutlery or chewing. If one wanted to identify the source of the recording, one would most likely answer that the recording is taking place in a kitchen (e.g., cooking, doing the dishes). In the above scenario, the person would take the task of auditory scene recognition. One of the goals of this research is to develop advanced methods that would enable a computer to do the same task, or as the scientific term refers to this problem as, ASR [10].

Practical applications of ASR include intelligent wearable devices [11], automatic speech recognition [12] and hearing aids that sense the environment of the user in real time [13]. The information about the environment enables the device to provide accurate and better service to the user. As mentioned above, ASR is used in content-based audio indexing and retrieval at the level of different environments.

The goal of acoustic event detection is to label temporal regions, such that each one represents a single event of a specific class. Early work in event detection treated the sound signal as monophonic, with only one event detectable at a time [14]. Events in a typical sound scene may be simultaneous; hence polyphonic event detection with overlapping event regions is desirable. There has been some work

on extending monophonic systems to polyphonic detection [15, 16]. Audio tagging is probably more demanding (complexity, computational cost) than monophonic event detection, but at the same time heavily intertwined to real-world scenarios where multiple audio events are co-occurring.

ASR can be jointly used with processing of other signals to obtain a more thorough understanding of user activities; for instance, processing of the fridge electrical power signal can also provide further information on a detected cooking activity, related to the times that the user opens the fridge and the time it is left open.

The main research question is how to improve the quality of living of elderly people with mild cognitive impairments using an unobtrusive sensor network, in order to understand their daily activities by creating a profile for each user (e.g., how many times there was a visit to the bathroom, time spent for cooking) and contact their relatives regarding abnormalities in their daily lives.

To answer this question there are several aspects that need to be covered. This research is going to cover the capabilities of individual sensor modalities and focus, specifically, on audio and energy consumption sensors. Robust audio event detection remains an unsolved problem. Environmental sound signals can vary according to the different environment (indoor, outdoor). Annotating audio data and assigning class labels (especially when sound events are overlapping) is a tedious task and since the human factor is involved, we can have mislabeling of the dataset. Secondly, background noise is a factor that affects the performance of an audio event detection system. Traditional approaches, such as filtering the input signal, can fail when the SNR is very small, meaning that we will "cancel" our entire input signal and miss any event for that time. Furthermore, another important question is, when applying deep learning models and our input data is not sufficient for training how we can explore additional data augmentation techniques for environmental signals. Regarding the energy consumption sensors, most researchers focus on data disaggregation (identifying appliances operating in a house from the total-main power consumption) techniques. Most public datasets (REDD [17], BERDS [18], Belkin Energy Dataset [19]) are released for research in novel methods for data disaggregation. Activity inference in a smart home environment from energy consumption has received less attention compared to the data disaggregation [20]. Finally, a thorough comparison of sensor modalities (energy, audio) in different house environments has to be conducted in order to understand which of these can output the highest accuracy for activity recognition. In theory the more types of sensors inside a house, the higher the accuracy

of specific activities. However, an important question lies with the recognition accuracy that can we achieve with one acoustic sensor placed at various distances from the target activities.

1.3 Project Objectives

Researchers are working on telemedicine and telemonitoring solutions to allow elderly people to stay at home as long as they can. Meanwhile, elderly care units have limited capacities (space and resource limitations). Mobility, which is currently a common request of companies, adds distance between family members. Elderly people often live alone and have to be autonomous. Moreover, with the increase of life expectancy, cognitive impairments such as dementia and diseases such as Alzheimer's are more and more prevalent. All this leads to implementing telemonitoring systems, able to detect a distress situation, or a significant change in the habits or behavior of the person.

Systems for enhancing medical diagnosis and information technology often focus on the clinical environment and depend on the extensive infrastructure present in traditional health care settings. Most high-fidelity sensor networks are expensive and require specialized training to operate [21].

Detecting normal and abnormal activities in a domestic environment necessitates an "always-on" and unobtrusive monitoring system. Gietzelt et al. [22] evaluated gait parameters measured by a single waist mounted accelerometer during everyday life of patients with dementia. Marschollek et al. [23] developed an unobtrusive method to determine individual fall risk based on the use of motion sensor data. Palmerini et al. [24] measured the acceleration of the low back to differentiate gait patterns in healthy adults and those with Parkinson's disease. An important scope of this research is to infer events offline, where no data will be sent to a cloud server but instead run locally on a Raspberry Pi, based on the input data. Most existing systems entail delays in detecting and reporting urgent situations. As such, offline activity recognition models are necessary for real-time analysis of sensor data. Therefore, it is important for a home care system to summarize the health status and daily behavior of the elderly, in order to guide the elderly towards healthy and active living. For instance, an activity recognition system would detect the activity of cooking for a week and notify the user's relatives with a phone call if no cooking activity was detected within the following week.

The innovative sensing infrastructure of this research aims towards the accurate, unobtrusive and privacy preserving monitoring of behavioral parameters and risk indicators in the daily living environment of older individuals and their use for the assessment of the mental condition of older individuals. An IoT environment comprising active power consumption and acoustic sensors will be the basis for retrieving large-scale data without posing any threats to the privacy of users. This data will then be processed and annotated to knowledge concerning the mental health and safety of their users, protecting at the same time their privacy and health record, using modern hardware and software data encryption approaches of the project's security and privacy framework. Specifically the recorded datasets (kitchen audio and power consumption) will be pseudo-labeled, the training would be done on a desktop computer and the inference on a Raspberry Pi would not save audio clips. Instead an audio stream will be stored in a buffer and logs of the recognized activities will be stored in a text file.

The main novelty of this research lies in utilizing the scope of different sensor modalities, using acoustic sensors and plug-wise energy meters. Specifically:

- The performance of each modality will be examined along with statistical significance tests. Starting from single-user single-activity use scenarios to real-world complex use scenarios, e.g., multiple users, interleaved and concurrent activity recognition
- Data mining and event detection recognition algorithms will be improved
- Feature extraction techniques and importance of CNN hyper-parameters in recognition accuracy will be explored
- A smart audio sensing infrastructure (integrated to a single board computer) using acoustic sensors will be created
- Deep learning algorithms for separating the speech from any other non-speech signals will be investigated

1.4 Proposed Approaches

This thesis presents the use of various machine learning and deep learning algorithms for the problem of human activity detection in domestic environments to

address the aforementioned aims and objectives. In order to achieve this an extensive review of the literature is carried out. Beginning the literature review on audio-based event detection from the theory of the computational auditory scene analysis to the machine and deep learning methods applied for the problem. The key relevant research in both audio-based event detection and speech/non-speech activity detection is surveyed. An understanding of the state-of-the-art approaches is provided and the research gaps are highlighted. Furthermore, the same process is followed for the energy-based event detection part.

Based on this understanding, two main frameworks are proposed to satisfy our objectives, about the audio-based event detection, set out to achieve. Using these frameworks, several tests are implemented to evaluate their performance in the task of audio-based event classification and speech activity detection. Various deep learning architectures are experimented to optimize our final system for high classification performance. For evaluation purposes tests with different parameters that are affected by randomness are conducted. Additionally, the filter size and the kernel size of the 2D CNNs and test their impact on classification accuracy are examined. Finally, the adaptation of the 2D CNN system in a natural language processing problem is examined and compared with traditional 1D CNN systems.

Regarding the energy-based event detection, a metric for the status of electrical appliances and the relation with the human activity is defined, Monte Carlo simulations and grid search are performed, in order to find the optimal parameter settings of traditional classifiers.

1.5 Contributions and Paper Publications

This thesis studies the task of using machine learning algorithms for human activity detection in domestic environments, with a particular focus on acoustic and energy consumption sensors. The contributions of this thesis are summarized as follows:

- Two frameworks for audio-based event detection are developed and statistical significance between traditional classifiers and a CNN architectures is examined.

- The statistical significance of well-known audio features for the problem of audio-based event detection in a kitchen environment, in the presence of background noise is examined.
- The effect of the duration of the signal segment used for classification on recognition accuracy is demonstrated.
- The SNR, the distance between the microphone and the effect of the audio source on the recognition accuracy in a new environment (i.e., one which was not used to train the classifier) is investigated.
- An end-to-end 1D CRNN and a 2D CRNN, using the 2D magnitude STFT representation as input, for speech/non-speech activity detection is proposed.
- A framework for unobtrusive human activity context inference, based on energy consumption rate from selected appliances and using a decision engine based on operation of the appliances is presented.
- The performance of CNN extracted features is studied. The experiments focus on comparison of automatically extracted and HCF for activity recognition. In particular, the audio signal, accelerometer and gyroscope data have been investigated. Moreover, the effect of important parameters, namely number of convolutional layers, and kernel size used for the convolutions, is evaluated.
- A framework for an NLP task based on visual features of text is presented. Similarly to understanding patterns in the magnitude representation of the audio spectrogram, two-dimensional CNNs, that use an image of a text as input, can build semantic representations which let them detect abnormalities in a text (i.e., garbage characters) without the need of OCR. This framework could help in the creation of a virtual assistant that would receive an image related to the timestamp and the activity detected by the audio sensor and perform a conversation between the user and the assistant.

During this PhD study, the following publications were made:

1. Federico Cruciani, **Anastasios Vafeiadis**, Chris Nugent, Ian Cleland, Paul Mc-Cullagh, Konstantinos Votis, Dimitrios Giakoumis, Dimitrios Tzovaras, Liming Chen and Raouf Hamzaoui. "Feature learning for Human Activity Recognition using Convolutional Neural Networks", CCF Transactions on Pervasive Computing and Interaction, 2020,

https://doi.org/10.1007/s42486-020-00026-2.
Contribution with respect to co-authors: Developed and implemented the system using the audio modality for the DCASE 2017 Task 1 development dataset and the ExtraSensory dataset.

2. **Anastasios Vafeiadis**, Konstantinos Votis, Dimitrios Giakoumis, Dimitrios Tzovaras, Liming Chen and Raouf Hamzaoui. "Audio Content Analysis for Unobtrusive Event Detection in Smart Homes", Engineering Applications of Artificial Intelligence, Elsevier, Vol. 89 (103226), 2020, https://doi.org/10.1016/j.engappai.2019.08.020.
Contribution with respect to co-authors: Data collection, designed and implemented the proposed frameworks.

3. **Anastasios Vafeiadis**, Eleftherios Fanioudakis, Ilyas Potamitis, Konstantinos Votis, Dimitrios Giakoumis, Dimitrios Tzovaras, Liming Chen and Raouf Hamzaoui. "Two-Dimensional Convolutional Recurrent Neural Networks for Speech Activity Detection", in Proc. 20th Annual Conference of the International Speech Communication Association (INTERSPEECH 2019), Graz, Sep.2019. ***Contribution with respect to co-authors***: The end-to-end convolutional recurrent neural network and the spectrogram 2D CNN for speech activity detection.

4. Federico Cruciani, **Anastasios Vafeiadis**, Chris Nugent, Ian Cleland, Paul Mc-Cullagh, Konstantinos Votis, Dimitrios Giakoumis, Dimitrios Tzovaras, Liming Chen and Raouf Hamzaoui. "Comparing CNN and human crafted features for human activity recognition", in Proc. 16th IEEE International Conference on Ubiquitous Intelligence and Computing (UIC), Leicester, Aug. 2019. (best student paper award) ***Contribution with respect to co-authors***: Developed and implemented the comparison systems for the audio modality using the DCASE 2017 Task 1 development dataset.

5. Erinç Merdivan, **Anastasios Vafeiadis**, Dimitrios Kalatzis, Sten Hanke, Johannes Kropf, Konstantinos Votis, Dimitrios Giakoumis, Dimitrios Tzovaras, Liming Chen, Raouf Hamzaoui and Matthieu Geist. "Image-based text classification using 2D convolutional neural networks", in Proc. IEEE Smart World Congress 2019, Leicester, Aug. 2019. ***Contribution with respect to co-authors***: Domain adaptation of the 2D CNN, using 2D magnitude spectrogram representation for audio-based event detection, to a natural language processing task.

6. **Anastasios Vafeiadis**, Thanasis Vafeiadis, Stelios Zikos, Stelios Krinidis, Konstantinos Votis, Dimitrios Giakoumis, Dimosthenis Ioannidis, Dimitrios Tzovaras, Liming Chen and Raouf Hamzaoui. "Energy-based decision engine for household human activity recognition", in Proc. Pervasive Computing and Communications Workshops (PerCom Workshops), 2018 IEEE International Conference on Pervasive Computing Athens, Mar. 2018. ***Contribution with respect to co-authors***: Development of the communication infrastructure for data collection, decision framework implementation.

7. **Anastasios Vafeiadis**, Dimitrios Kalatzis, Konstantinos Votis, Dimitrios Giakoumis, Dimitrios Tzovaras, Liming Chen and Raouf Hamzaoui. "Acoustic Scene Classification: From a Hybrid Classifier to Deep Learning", in Proc. Detection and Classification of Acoustic Scenes and Events 2017 Workshop (DCASE 2017), Munich, Nov. 2017. ***Contribution with respect to co-authors***: Design and implementation of the hybrid SVM-HMM and 2D CNN systems.

8. **Anastasios Vafeiadis**, Konstantinos Votis, Dimitrios Giakoumis, Dimitrios Tzovaras, Liming Chen and Raouf Hamzaoui. "Audio-based event recognition system for smart homes", in Proc. 14th IEEE International Conference on Ubiquitous Intelligence and Computing (UIC), San Francisco, Aug. 2017. ***Contribution with respect to co-authors***: Data collection and design of the system based on traditional classifiers.

1.6 Outline of the Thesis

The thesis is organized as follows. Chapter 2 presents the background of HAR, with an emphasis on sound and energy-based activity recognition and the privacy issues related to each modality. Additionally, it presents related work on human crafted engineering features, focusing on IMUs and acoustic sensors, and the features learnt by a CNN. The aim of this Chapter is to identify the research gaps in the literature.

Chapter 3 describes the approach in the area of acoustic event classification in smart homes and presents two approaches for audio-based event detection in an indoor environment. Moreover, several advanced classification techniques are compared in that Chapter. Statistical importance of audio features and classifiers are also investigated, applying McNemar's test. The proposed frameworks are tested

for their generalization ability in an environment that is not seen during training and their ability to recognize a class not included in the training, but with similar sound characteristics as the trained ones.

Chapter 4 details the problem of the speech activity detection and propose two frameworks. An end-to-end one dimensional CRNN and a two dimensional CRNN. The Chapter highlights the importance of the post-processing convolution operations for the problem of speech activity detection and their effect in a real-world dataset.

Chapter 5 describes the work done in the energy-based activity detection problem, using a decision engine based on active power consumption. There is a focus on the data collection infrastructure and the classification accuracy of traditional classifiers is compared, using the active power consumption as the only feature vector.

Chapter 6 presents the study of human crafted features and the automatic CNN features in real-world datasets. The effect of the number of layers and the kernel sizes of simple CNN architectures is compared. The second part of the Chapter presents the adaptation of a 2D CNN, which was used for audio-based event detection, in a NLP problem. It shows that for a dialogue task the 2D CNN, which receives an image of a text as input, can significantly outperform the memory networks without a match type.

Finally, Chapter 7 concludes the thesis, summarizing the contributions, the limitations of this work and presenting the open issues for future work.

Chapter 2

Background

2.1 Introduction to Human Activity Recognition

Activity recognition methods in smart homes rely mostly on sensors, which are further separated into wearable [25] and environment-related ones [26]. Recent work [27] shows that ontologies and semantic technologies have been used for activity modeling and representation. Wearable-based techniques depend on user interaction with the sensor and in most cases on user motion by employing accelerometers.

He et al. [28] provide the time series from such sensors as input to autoregressive models to extract features and classify four activities (running, still, jumping and walking) utilizing an SVM classifier. Subsequent works [29, 30] use more advanced models to extract features, namely autoregressive combined with signal magnitude area and tilt angles, while employing modules to further enhance the data separability, for activities such as resting (lying/sitting/standing), walking, walking-upstairs, walking-downstairs, running, cycling, and vacuuming. Plötz et al. [31] perform feature extraction on the time-series using layered Restricted Boltzmann Machines, on four public datasets. Considering methodologies focusing on sensors attached to the environment, there is a significant diversity of types, ranging from light sensors, humidity ones, thermometers and others. Since the extracted information from such sensors can be insufficient, they are usually accompanied by accelerometers [32], which are attached to objects of interest. However, this category of methods fails when activities do not involve the registered objects. Several

works [33, 34] place sensors on objects and track their movement, thus activities are inferred through the traces of the objects which are modeled using a HMM or a DBN, respectively. Bourobou et al. [35] propose a two-step procedure that relies on smart environments to recognize five activities (taking a bath, preparing breakfast, listening to music, playing a game and preparing lunch). The first step comprises the discretization of activity patterns while the second step trains an ANN based on the temporal relations of those patterns.

Smart home-based ambient assisted living ICT solutions can allow the elderly to remain in their own homes and live independently for longer [1]. Research on ICT solutions for ambient assisted living has intensified over the last decades considerably, due to the emergence of affordable powerful sensors and progress in artificial intelligence [25, 36, 37].

Various HAR systems that monitor daily activities to identify abnormal behavior have been proposed for ambient assisted living applications [32, 38].

One common approach to automated HAR uses portable sensors such as accelerometers and gyroscopes [39, 40]. However, these sensors require cooperation of the subject, may restrict body movement, and are energy constrained [41, 42]. Another approach relies on computer vision [43, 44]. However, privacy concerns are hindering its adoption. A further approach is based on audio processing. Features are extracted from the environmental sounds and classifiers are used to recognize the corresponding human activity [45, 46, 47].

2.2 Audio-based Event Detection and Activity Recognition

Audio-based activity recognition has received a lot of attention from researchers in recent years [48, 49]. A number of studies have also taken the first steps to characterize the indoor sound environment and the classification of events [50, 51].

While many approaches addressed the problem of audio-based activity recognition in a home environment [52, 53, 54, 55], there is not enough justification for the classifier and feature selection. Most of them used well-known features from the field of speech recognition (e.g., MFCCs) along with classifiers, such as the kNN algorithm, to serve as a proof of concept for indoor audio-based activity classification. Chu et al. [56] showed that increasing the number of audio features does not

improve the recognition accuracy of a system classifying environmental sounds and used the matching pursuit algorithm to obtain effective time-frequency features.

DNNs are able to extract important information from the raw data without the need for hand-crafted feature extraction and outperform traditional classifiers in many tasks. There is significant research on recognizing single events in monophonic recordings [57] and multiple concurrent events in polyphonic recordings [58]. Different feature extraction techniques [59], hybrid classifiers [60, 61] and very deep neural models [62] have been explored. However, none of these works compared 1D and 2D CNN architectures for ambient sounds.

Another focus of this work is the duration of the signal used with an audio-based event detection system. The works [63, 64, 65] examined the length needed for sufficient recognition accuracy. They used systems based on time-frequency features and simple classifiers, such as SVMs and HMMs. The proposed approaches work well with datasets that contain indoor or outdoor environmental sounds. However, due to the high variability in the class and the similarity between different classes, they can fail in a specific acoustic event detection task (e.g., in a kitchen environment).

Finally, there has been extensive research on the effect of the SNR in the presence of background noise [66, 67, 68]. Wang et al. [69] performed experiments for various artificially added SNRs (0-10 dB and clean recordings) using different environmental sound datasets and a hierarchical-diving deep belief network. However, all previous work assumed prior knowledge of the SNR, which is not possible in a real-world environment.

2.2.1 Computational Auditory Scene Analysis

CASA is the study of auditory scene analysis by computational means. In essence, it refers to how the human auditory system organizes the sounds of a complex environment. Bregman [70] was one of the pioneers in this field, studying the processes in the human brain, how humans perceive the auditory environment by grouping sounds into objects. Bregman suggests that there are many phenomena going on in the auditory perception that are similar to those in visual perception, such as exclusive allocation (properties belonging to one event only) and apparent motion (a motion is perceived, although the stimulus is not moving). Dan Ellis has contributed a lot to the research on CASA. In his Ph.D. thesis [71] he presents an

approach, in which the analysis is done by matching the predictions of an internal world model and observed acoustic features.

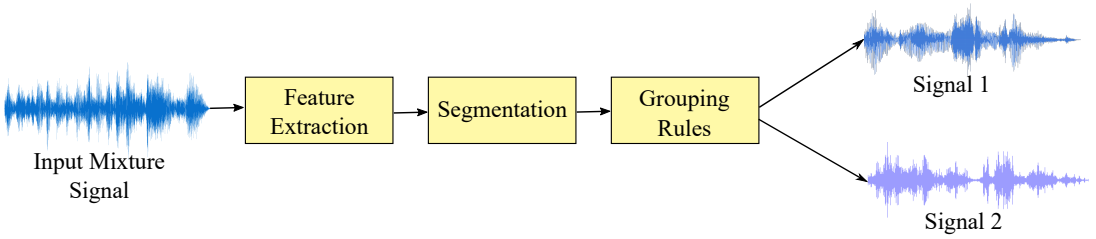


FIGURE 2.1: CASA System Overview

Figure 2.1 shows a typical architecture of a CASA system. All these systems start with an analysis of the signal in the time-frequency domain. In most cases, a gammatone filterbank [72] is applied to the input signal, in order to "mimic" the frequency selectivity of the human ear.

The next stage includes the feature extraction of the signal. Some of the most important features are the fundamental frequency, onset/offset of the signal, amplitude/frequency modulation. The extracted features enable the system to form the segments. The segments provide a mid-level representation on which grouping operates. Finally, grouping rules are applied, in order to identify components that have the highest probability to come from the same source [73].

2.2.2 Acoustic Scene Recognition

ASR is closely related to CASA. ASR is a particular task that is related to CASA, however, the focus is the context recognition, or the environment recognition, rather than the analysis and interpretation of discrete sound events [74]. Applications of ASR include intelligent wearable devices and hearing aids that sense the environment and adjust the mode of operation accordingly.

Research on unstructured audio recognition, such as environmental sounds, has received less attention than that for structured audio such as speech or music. Only a few studies have been reported, and most of them were conducted with raw environment audio. Because of randomness, high variance and other difficulties associated with environmental sounds, their recognition rates are smaller than those for structured audio are. This is especially true when the number of sound classes increases. To overcome the insufficiency of MFCCs and other commonly used features, Chu et al. [56] proposed a set of features based on the MP technique.

Although the MP-based features provide good performance, their computational complexity is too high for real-time applications.

The performance of ASR algorithms dramatically decreases when the number of sound classes increases (even with good features). It will need more good features for performance improvement. On the other hand, a larger number of features not only results in higher complexity but also demands more samples while there is no guarantee on performance improvement. That is because some features may help classify some classes but hamper the classification results for the others. Besides, it is not easy to train a classifier for better discriminant power in a higher dimension feature space. As a result, feature selection and reduction is an important task.

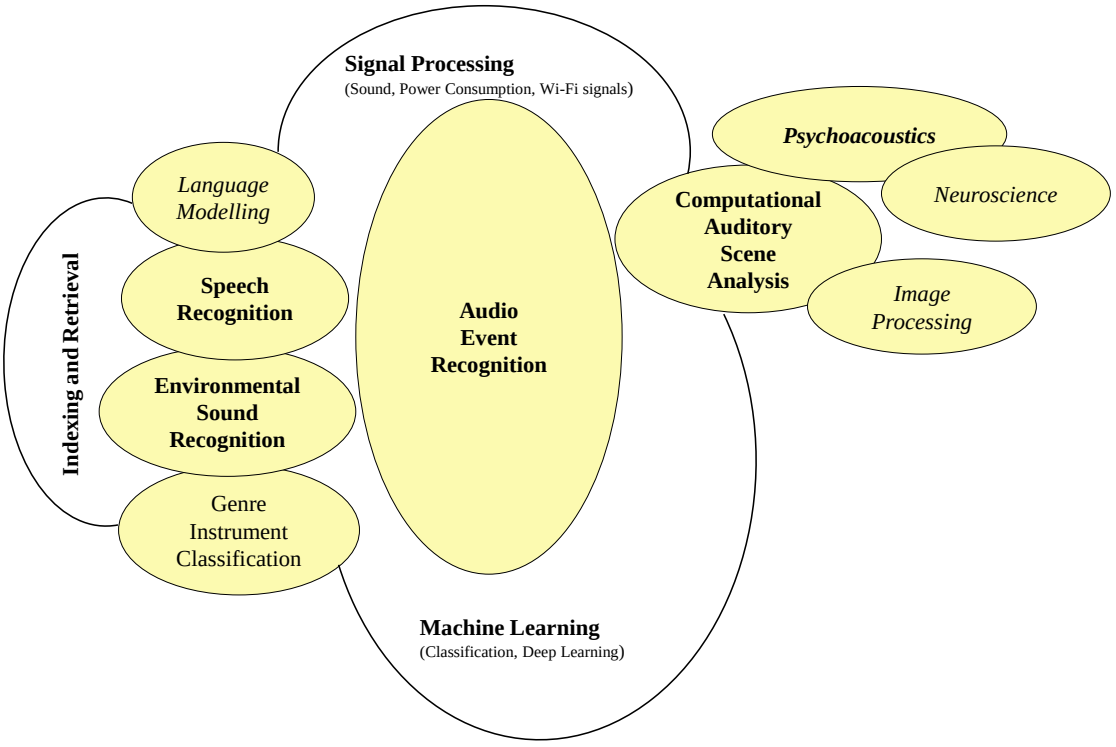


FIGURE 2.2: Intersection of Audio Event Recognition with the wider scientific field

Figure 2.2 shows how the field of audio-based event detection intersects with other scientific fields, such as psychoacoustics (e.g., ERB scale for the construction of the GFCC features) and how environmental sound classification is often based on methods from the voice activity detection and music genre classification. Examples of domain intersection include data augmentation techniques (e.g., image rotation, salt and pepper noise) that are applied in the image domain but also in the audio/spectrogram respresentation, DNNs designed for computer vision problems that are also applied to the audio-based event recognition problem, audio

features that are important for speech and music classification, can be extracted for environmental sound classification.

2.2.3 Deep Learning in Audio-based Event Detection

Deep learning is a machine learning field based on learning direct representations from the input data rather than task specific algorithms (e.g., time-dependent algorithms such as HMMs and time-independent algorithms such as Random Forests). The first neural network can be tracked back in 1967 [75], when Rosenblatt introduced the idea of a perceptron unit capable of learning weights from input data to find patterns. The ability of deep learning networks to extract unique features from raw data and the high processing speeds of modern GPUs lead these networks to receive a lot of attention in a wide range of scientific fields, such as natural language processing, image/video classification and segmentation, reinforcement learning and audio event detection.

Lee et al. [76] were one of the first ones to introduce unsupervised learning for audio data using convolutional deep belief networks. In particular they showed that the learned features from the neural networks corresponded to phones/phonemes in speech data. They also showed that these models could be applied to other datasets, such as music genre classification with promising results (comparing to traditional MFCC extraction with a classifier). Since then there were a number of research outcomes in the field of speech recognition [77, 78, 79, 80].

Piczak [81] tested a very simple CNN architecture with environmental audio data and achieved accuracies comparable to state-of-the-art classifiers. Cakir et al. [82] used one dimensional (time domain) DNNs in polyphonic sound event detection for 61 classes to achieve an accuracy of 63.8%, which was a 19% improvement over a hybrid HMM/Non-negative Matrix Factorization method. Lane et al. [83] created a mobile application capable of performing very accurate speaker diarization and emotion recognition using deep learning. Recently, Wilkinson et al. [84] performed unsupervised separation of environmental noise sources adding artificial Gaussian noise to pre-labeled signals and used auto-encoders to cluster. However, background noise in an environmental signal is usually non-Gaussian, making this method to work on specific datasets only.

Creating a network for robust environmental sound classification in different environments with highly variable datasets (in terms of noise) remains an open issue. There is a strong need of a model that would generalize with unlabeled data.

2.2.4 Privacy Issues

The key question regarding acoustic monitoring for activity recognition is the following: Will sound "surveillance" be socially acceptable in private places, such as bedroom or bathroom where the use of video is not? Therefore, a framework has to be developed in order to address this issue.

A useful term about the privacy discussion is Palen and Dourish's genre of disclosures [85], in which they created social patterns of privacy management involving recognizable, socially meaningful patterns of information disclosure and use. Given a genre, people expect each other to disclose this information but not that, under these conditions but not those, to this but not that person, and to use information in this but not that way. Within this framework, the degree of perceived "privacy loss" caused by the introduction by a new technological construct is related to its non-conformity to these expectations of disclosures within a given genre.

2.3 Energy-based Event Detection and Activity Recognition

2.3.1 Statistical and Traditional Machine Learning Approaches in Energy-based Event Detection

Many approaches have been proposed to address the problem of activity recognition in domestic environments. Most methods operate on the basis of multi-parametric data, taken from multiple modalities; i.e., various kind of sensors installed in the house environment. Kim et al. [86] compared the performance of HMMs, CRFs and the Skip-Chain CRF, of eating activities in a home environment. Nazefrad et al. [87] compared the performance of HMMs and CRFs for activity recognition in a smart home environment, using real time data from motion and temperature sensors. Giakoumis et al. [43] proposed an activity recognition scheme for daily activities such as cooking, eating, dishwashing and watching TV, based on depth video processing and Hidden Conditional Random Fields, achieving an overall accuracy of 90.5% in a natural home environment.

More recently, Stankovic et al. [88] proposed an innovative methodology that characterizes the energy consumption rate of domestic life by making the linkages

between appliance end-use and activities through an ontology built from qualitative data about the household and NILM data. Lavin and Klabjan [89] proposed a clustering technique for time series, on energy usage data provided by several U.S. power utilities, aiming to compare and contrast those with similar energy usage tendencies and to identify potentials for energy efficiency along with the open and close hours for business.

Cottone et al. [90] trained an HMM as an automated activity recognizer using sensors network readings initially converted into meaningful events, by applying a lossy compression algorithm based on minimum description length. The aim of their work was to level out peaks of energy consumption rate by identifying the appliances whose service is effectively needed by users, and postponing the use of the others until the combined demand for energy falls below some predefined threshold. In the work of Xu et al. [91] an alternative scalable two- stage methodology for household consumption segmentation is proposed that considers both the shape of a load profile (the time and magnitude of its peaks in household appliances consumption) along with its overall consumption to determine different typical consumer behavior patterns. Rao et al. [92] proposed an approach combining machine learning (SVMs) edge analysis and time series models (autoregressive moving average) on the identification of active appliances and on the prediction of future power usage utilizing demographic data in addition to aggregate power usage over time. Deshmukh and Lohan [93] proposed a framework for creating the appropriate features and labels from the training data and used these features to predict the appliance status (ON/OFF) and appliance energy consumption rate using a variety of classifiers. Finally, Belley et al. [94] proposed an algorithm for human activity recognition extracting features from the active and reactive power of each device using a Gavazzi meter. This method does not consider the case where a smart home would be equipped with several devices of a specific model. Furthermore, it would require the purchase of materials to measure the harmonics in power system in some cases, such as the television.

Contemporary activity recognition methods in smart homes rely mostly on sensors, which are further separated into wearable [26] and environment-related ones [25]. Recent work [27] shows that ontologies and semantic technologies have been used for activity modeling and representation, as well as knowledge-driven approaches [95]. Wearable-based techniques depend on user interaction with the sensor and, in most cases, on user motion measured with accelerometers.

2.3.2 Deep Learning in Energy-based Event Detection

Most of the deep learning frameworks, developed for a NILM task, focus on disaggregating specific electrical appliances. That means that given only the total power consumption of a house, one can identify operating appliances.

Kelly et al. [96] developed three deep neural network architectures (feed-forward fully connected, recurrent neural network, one-dimensional autoencoder) for energy disaggregation of five appliances (kettle, washing machine, dishwasher, fridge and microwave), using a real-world dataset (UK-DALE). Zhang et al. [97] used a CNN architecture for sequence to sequence learning on the UK-DALE dataset and the same five appliances as Kelly et al. They showed that the sequence to sequence learning based on CNNs can achieve better results in disaggregating appliances, since the networks can capture most of the information in the time domain. Finally, Krystalakos et al. [98] pointed the importance of sliding windows in the time domain that could significantly affect the performance of the neural network architectures.

Despite the developed approaches, there is a strong need for reproducibility of the results reported in the papers, as with any deep learning algorithm. Additionally, for the NILM problem, there are appliances (e.g., desk lamp) that occupy a small portion of the total energy consumption and are therefore, not easily detectable. The architectures developed for a specific dataset may not work on another dataset and finally, any developed architecture should report the results on the same time windows, when using public datasets, in order to have a fair comparison.

2.3.3 Privacy Issues

Privacy using the active power consumption sensors is related to the energy data that are transmitted to an IoT device for further processing. A few services that an IoT device can provide are elderly monitoring, theft detection and management of the power consumption [99]. Most of the times there is a client/server based solution, either via a RESTful application or another telemetry transport protocol, the data is sent over the cloud. Despite the useful services that can be provided to the user, the can be accessed by attackers and important information about human activity can be retrieved. Past studies [100, 101] have focused in developing unsupervised, non-invasive privacy protection techniques based on public datasets from real-world homes.

2.4 Comparison of Human Crafted Engineering Features and Learnt Features of a CNN for Activity Recognition

HAR finds several real-life applications; in smart home research, for instance, it can be applied to support AAL [102][103]. AAL is among the application scenarios making this research branch particularly relevant. Its relevance is linked to the potential role AAL could play in dealing with rising healthcare costs associated with an ageing demographic [104]. At the same time, HAR is also among the main fields of application of ML. As in other cases of ML applications (e.g., speech-recognition or visual object recognition), DL has been increasingly employed in recent years, and its adoption has led to a significant improvement in the state-of-the-art performance metrics [105]. HAR is no exception in this sense. DL methods such as CNN and LSTM networks have shown good results in terms of recognition accuracy both in the case of simple activities (e.g., 'sitting', 'standing', 'walking') and complex activities (e.g., preparing a meal) [102, 106]. One of the distinctive traits of DL approaches to HAR is that these methods simplify some stages of the conventional approach taken to activity recognition, by automating some of the steps commonly employed in similar classification tasks (e.g., feature extraction). In sensor-based HAR, DL allows direct use of raw data as input [105] (e.g., in the case of HAR based on inertial sensors [107]). For instance, CNNs have been successfully employed to extract relevant features from the accelerometer raw data signal, in an automated fashion [102]. The ability of processing data in its raw form represents a disruptive change in comparison to conventional ML approaches. In the past, the step of feature extraction would require (in most cases) an advanced degree of domain specific expertise [105]. Despite the popularity of DL methods, very little focus has been put on automatically extracted features, particularly in comparison with the case of HCF. While in the first case, HCF have been the objective of several studies attempting to identify optimal feature sets for different target activities [108, 109], for automatically extracted features, implementation of the HAR chain focuses on the recognition accuracy performance of different classifiers and less on the impact of the extracted features. Some recent studies have attempted to fill this gap [102, 110]. In [102], 1D temporal convolution has been examined for HAR using inertial sensors. In [110], different feature learning methods have been compared including CNN, LSTM and HCF. The comparison, however, has been measured on the final F1-score obtained on the same dataset

using different methods, rather than focused on comparison between HCF and automatic features.

The process of activity recognition usually includes a specific sequence of steps, also known as ARC [111]. The chain describes the process of HAR going from raw data to final classification, and includes the following steps: pre-processing, segmentation, feature extraction, and classification. In the pre-processing step, raw data are processed in order to transform them into a form suitable for processing by a classification model. Typical operations performed at this stage include for instance filtering (e.g., in the attempt or removing noise or not relevant parts of the signal), or re-sampling, as in the case where multiple inputs acquired at different sampling rates need to be put together into a time-series. In the segmentation step, the data sequence is divided into a set of segments. This operation introduces a relevant parameter for classification which is the window size, that will determine the length of each segment [111]. The choice of the window size can influence the ability of extracting informative features from the segment. If, for instance, the window size is too short, relevant features may be missed. Window sizes of 1-4 seconds are often used for HAR of simple activities, while larger window sizes are considered generally for complex activities, as in the case of ADL [107, 112]. The following step in the ARC is the feature extraction step [111], that typically requires domain specific expertise [105]. It is common at this stage to perform an additional step, known as feature selection. Feature selection normally consists in an iterative process, in which different subsets of HCF at each iteration are evaluated based on the final accuracy. This process allows for the identification of an optimal set of features [113]. Although feature selection further exacerbates the complexity of generating a candidate feature set, this process can be automated, as for instance in [113]. In this case, an algorithm has been proposed to discard features with low importance (i.e., not improving classification accuracy), however, the initial set of features prior to selection are still human crafted. DL approaches facilitate automation of both feature extraction and selection [105], and that is also the case of CNN based methods [102, 103, 110] but do not omit the entire ARC. In this work, two main cases are analyzed: inertial sensors and audio signals for HAR. Consequently, the following subsections will describe common feature extraction techniques for the two cases.

2.4.1 Inertial sensor

Inertial sensors are commonly used in HAR as they are less power demanding compared to other sensors, such as GPS, and do not pose privacy issues by capturing location data anonymously [114], which occur in the case of video-based approaches. Several studies have investigated different HCF sets as well as feature selection techniques [108]. Common features used for inertial-based HAR usually belong to two main groups, depending on the fact if they are extracted from the time or the frequency domain [107, 108]. Time domain features are more often used and include the statistical moments of the signal (e.g., mean, variance, skewness), or other simple features such as max and min values in the interval. Frequency domain features require DFT computation, and therefore are more rarely employed due to the inherent computational complexity [107]. Identifying an optimal feature set for HAR has been the objective of several studies. Consequently, it is possible to rely on existing literature, that provides a comprehensive analysis of feature quality, as in [108]. The UCI-HAR dataset [115] includes a set of 561 features extracted from accelerometer and gyroscope signals (348 considering the accelerometer only), both from the time and frequency domain and both at single axis and at magnitude level. When focusing on DL approaches, and particularly on automating feature extraction, only few studies attempted to do an analysis of produced features [103]. Feature learning strategies including DL has been the objective of [110]. In this case however, the comparison is provided only on the final accuracy of models, that are trained using different approaches (including CNN). In [102], a more detailed analysis of features extracted using CNN is provided, including an insight on the effect of main parameters used for CNN classifiers (e.g., number of convolutional layers and filter size). The objective of the study, however, was to evaluate optimal CNN parameters for HAR, rather than focusing on CNN features. Also in this case, results were provided on the final accuracy of the CNN approach, measured on the UCI-HAR dataset [115]. Moreover, the study analyzed only the case of combined accelerometer and gyroscope signals, while in this work, results obtained using both combined accelerometer and gyroscope, and accelerometer only, are presented.

2.4.2 Audio Features

In the field of computational auditory scene recognition, feature extraction remains a fundamental problem. Many types of low-level features such as zero-crossing

rate, band-energy ratio, spectral roll-off, spectral flux, spectral centroid, spectral contrast, MFCCs and gammatone frequency cepstral coefficients are commonly used in the literature [10, 64, 116, 117]. The majority of the features selected within these studies, however, only work well for structured data, such as speech and non-speech separation or music genre classification. Therefore, a more discriminative feature set that captures the spatial and temporal events is required, especially for environmental sounds. Recently, deep CNNs have been successful in many tasks such as, speech recognition [118], audio source separation [119] and environmental sound recognition [120]. However, the problem of audio-based event recognition remains a hard task. This is because DL approaches that work extremely well for a specific dataset may fail for another. The fundamental difficulty of environmental sound recognition is that the input signal is highly variable due to different environments (indoor, outdoor, vehicle) and acoustic conditions.

Chapter 3

Audio-based Event Detection

3.1 Introduction

While several audio-based HAR systems have been proposed, a number of important questions remain unanswered:

- which features and which classifiers are most suitable in the presence of background noise?
- what is the effect of the duration of the signal segment used for classification on recognition accuracy? Decreasing the segment duration decreases the response time of the system but may harm its recognition accuracy. At the same time, increased duration can lead to increased co-occurrence of multiple events within the same sound segment;
- how do the SNR and the distance between the microphone and audio source affect the recognition accuracy in a new environment (i.e., one which was not used to train the classifier)?

This work answers these questions for a real-world indoor kitchen environment where large audio datasets are captured and processed to train classifiers. Two representative AED approaches are studied. The first one extracts time and frequency features and uses a traditional classifier. We compared various features and classifiers and showed that the best results are obtained with hybrid time-frequency features, together with a gradient boosting classifier. Our best system achieved an F1-score of 90.2% and a recognition accuracy of 91.7%. The second system uses the two-dimensional mel-spectrogram magnitude representation

of the audio signals as input to a 2D CNN. We showed that compared to a 1D CNN that applies max-pooling to only one dimension, applying max pooling to both dimensions of the input (time and frequency) reduces the dimensionality in a more uniform manner, yielding more salient features with each consecutive convolutional-max pooling operation. Although 2D CNNs usually outperform 1D CNNs in many tasks [121], when deploying the network in a system-on-chip device, one has to consider the complexity of the network, the number of network trainable parameters and the effect on the recognition accuracy. The proposed 2D CNN achieved a recognition accuracy of 96% and an F1-Score of 92.7%, while maintaining a small number of trainable parameters (approximately 550,000). Additionally, we observed that in a real-world environment the recognition accuracy for some classes did not improve when the signal duration was greater than 3 s. This was due to overlapping sounds that occurred in the kitchen environment (e.g., kitchen faucet running, while the user picks a plate to wash). Even in the cases where the recognition accuracy increased, the improvement was not significant. Studying the trade-off between signal duration and accuracy is important in scenarios where the data needs to be captured and processed on a system-on-chip device (e.g., Raspberry Pi), with limited memory size. Finally, since real-world environments typically include noise, we studied the effect of the SNR and distance between the microphone and the target audio event on the recognition accuracy. For events such as using the mixer and the utensils (forks, spoons, knives), the recognition accuracy was high despite the background noise of a kitchen fan and a refrigerator. The high amplitudes in the signals associated with these events could mask the low amplitudes of the background noise signals. On the other hand, we noticed a drop in the recognition accuracy for quieter sounds (e.g., dishwasher). We did not add artificial background noise to affect the SNR since we wanted to be as close to a real-world scenario as possible. The classification results that we obtained at various distances showed that we can achieve good accuracies with one microphone. This was useful, especially for monitoring houses of the elderly, where the number of sensors should be as small as possible.

The two systems are unobtrusive and preserve privacy as the raw audio is immediately deleted after feature extraction and cannot be recovered from the features.

The rest of the Chapter is organized as follows. Section 3.2 describes the two systems used in our study, giving details on signal acquisition, feature extraction, feature selection and classification. The results are presented in Section 3.3. Finally, Section 3.4 concludes this Chapter.

3.2 Acoustic Event Detection Frameworks

We propose two approaches for acoustic event detection in an indoor environment.

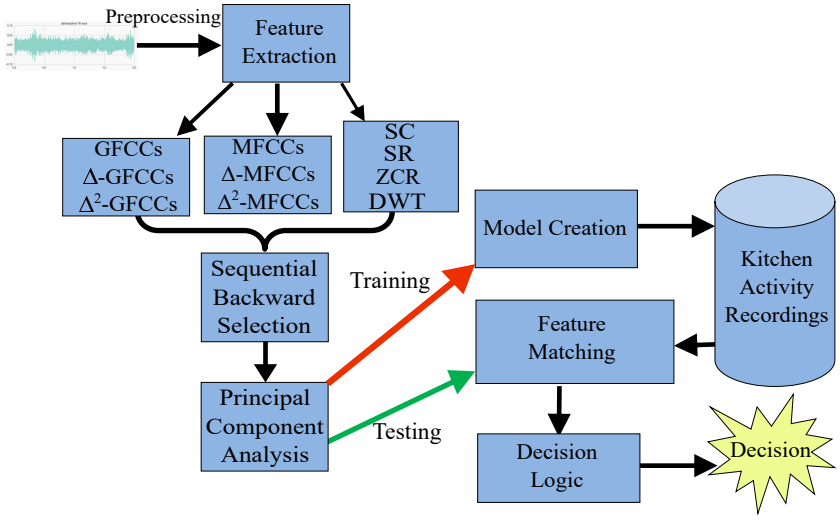


FIGURE 3.1: First proposed AED approach.

For the first AED approach (Figure 3.1), we considered time-domain features (ZCR), frequency-domain features (MFCCs, GFCCs, SR, SC), and time-frequency features (DWT coefficients). Furthermore, we studied the effect of adding many audio features along with proper feature selection and reduction techniques on recognition accuracy. For classification, we examined well-known classifiers such as kNN, SVM, Random Forest, Extra Trees and Gradient Boosting [122].

For the second AED approach, we used a CNN trained on mel-spectrogram images (Figure 3.6). We show that even for a small dataset, a 2-dimensional CNN with 2-dimensional max-pooling (downsamping) layers can provide good recognition accuracy results. The details of the two approaches are given in the following sub-sections.

3.2.1 Signal Acquisition

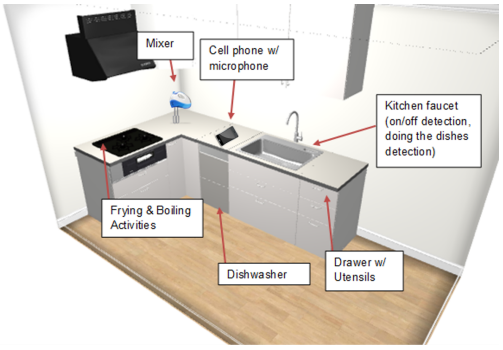
The success of the signal recording depends on the environment and the placement of the microphone. Ideally the recordings should take place in soundproof studios or labs. However, this is not possible in real life. Therefore, we examined test case scenarios with various types of noises that may occur in a home environment.

Three kitchen environments (author's house, CERTH's nZEB smart home and AKTIOS S.A. Elderly Care Units in Vari, Athens [1]) were used for data collection.

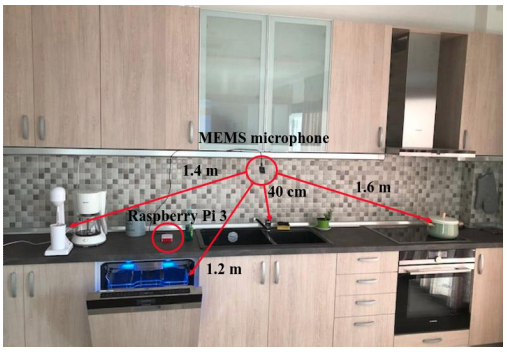
In the first step of the preprocessing, we recorded the input signal in stereo at 44,100 Hz (16-bit depth) and then averaged the two channels. This allowed us to use frequencies up to 22,050 Hz, according to the Nyquist criterion. This is sufficient to cover all the harmonics generated by our input signal and removes noise above this range (also not detected by human ear).

Sounds of activities using the kitchen setup of Figure 3.2 (a), were recorded, where there was no background noise and Figure 3.2 (b, c) that included background speech sounds and ambient noise of a fan and refrigerator. Sounds for the seven classes were collected from Freesound [123].

(A) Home Kitchen Environment Setup



(B) CERTH Smart Home nZEB setup



(C) AKTIOS Kitchen Environment Setup

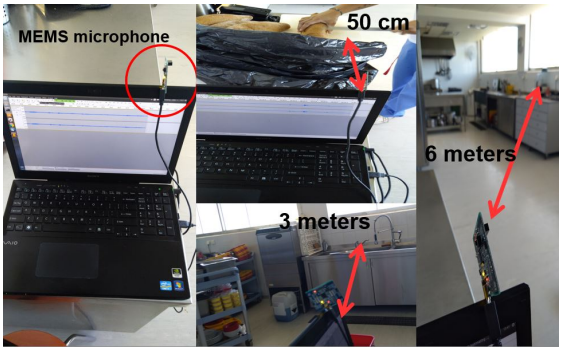


FIGURE 3.2: Experimental Setup

The first recordings were made in the kitchen of the first author (Figure 3.2 (a)). Only one person was present at the time of the recordings.For this environment, two smartphones (Samsung Galaxy S5 & ZTE Nubia Z11 miniS) were placed on the kitchen counter above the dishwasher at an identical position. The main reason

[1]Ethical approval at Appendix A

for using two smartphones was to capture the same source from two different, off the shelf, microphones. The smartphones were 50 cm away from the faucet, approximately 50 cm from the mixer, 1 m from the oven and approximately 2 m from the kitchen drawer. For the second set of recordings, the setup was as follows (Figure 3.2 (b)):

1. we used a Raspberry Pi 3 Model B with an MEMS DSP board to record the audio signals
2. the environment was noisier than for the first set of recordings because other researchers were present and speech or other environmental noises were captured more frequently
3. the MEMS board was placed at 1.2 m from the dishwasher, 1.4 m from the mixer, 40 cm from the kitchen faucet and 1.6 m from the oven. Compared to the first recordings, the distances to the appliances were larger to reduce the SNR

Finally, for the third set of recordings (Figure 3.2 (c)), we classified the audio signals in near real-time (approximately 450 ms delay) using a laptop and an MEMS microphone board. For this environment there was a background noise of a fan and a refrigerator. We used the MEMS board to manually adjust the microphone gain (+6 dB; maximum threshold to avoid clipping when placed within 50 cm from the cutting board to detect the activity of bread cutting) for the recordings and the laptop to perform near real-time classification. The MEMS board was placed in a fixed position on top of the laptop and 3 m from the kitchen faucet, 3 m from the dishwasher, 6 m from the mixer, and 50 cm from the cutting board.

TABLE 3.1: Number of recordings of each class from different sources

Classes	**Kitchen Environment Figure 6(a)**	**Kitchen Environment Figure 6(b)**	**Kitchen Environment Figure 6(c)**	**Freesound**
Frying	160	85	-	40
Boiling	160	85	-	40
Mixer	160	40	45	40
Doing the dishes	160	85	-	40
Kitchen sink	160	34	51	40
Dishwasher	160	20	65	40
Cutting bread	-	-	285	-

A total of 1,995 audio signals from different activities were collected from the three kitchen environments (285 kitchen faucet, 285 boiling, 285 frying, 285 dishwasher,

285 mixer, 285 doing dishes and 285 cutting bread). All signals had a duration of 5 s. The dataset was manually labeled by doing one event at a time. The setup included the following steps:

- we used data augmentation techniques as described in Section 3.2.2 to increase the total number of recordings in each class to 855
- Monte Carlo cross-validation was used to randomly split the dataset into training and testing data (80% training and 20% testing) and the results (accuracy, precision, recall, F1-score) were averaged over the splits

The number of recordings for each class is summarized in Table 3.1.

3.2.2 Data Augmentation

Environmental audio recordings have various temporal properties. Therefore, we need to make sure that we have captured all the significant information of the signal in both the time and frequency domain. Any environmental signal is a non-stationary signal [56], since it is a stochastic signal and a signal value is not equally probable to occur given another signal value at any time instance.

Previous research [124, 125] showed that data augmentation can significantly improve the performance of a classification system by introducing variability into the original recordings. For this reason, for both AED approaches, we produced two additional recordings from the original ones. First, for each recording, we added white noise with uniform probability distribution. This allowed us to train our system better, since the test audio data in an unknown environment (not used for training) would also include various noises (e.g., different people speaking while performing an activity such as cooking). Second, we re-sampled the original recording from 44.1 kHz to 16 kHz. Most of the monitored kitchen environment recordings (mixer, dishwasher, faucet, utensils) had a fundamental frequency of around 600-700 Hz. We focused on the harmonics produced by devices such as the mixer and the dishwasher and found that a lot of information at around 11 kHz was necessary for these classes.

The quality of the data was maintained since i) downsampling removed the frequencies above 16 kHz and did not affect the general recording since the energy of the highest frequencies (above 16 kHz) was very small and ii) the added uniform

noise corresponded to the scenario where ambient noise was present in the kitchen environment (e.g., fan and refrigerator of the setup in Figure 3.2 (c)).

3.2.3 Feature Extraction

To include the range of frequencies that are relevant to identifying the kitchen environmental sounds and to efficiently extract the audio features, we split the input signal into smaller frames for processing. Each frame had a window size of 20 ms with a 10 ms hop size from the next one (50% overlapping sliding Hamming window). Thus, there were 173 frames per recording.

For the second AED approach, we calculated the mel-spectrogram with 128 bins to keep the spectral characteristics of the audio signal while greatly reducing the feature dimension. We normalized the values before using them as an input into the CNN by subtracting the mean and dividing by the standard deviation.

In the following, we give the details of feature extraction for the first AED approach.

3.2.3.1 Mel-Frequency Cepstral Coefficients

MFCCs are one of the most popular features for voice recognition [126]. Figure 3.3 shows the steps involved in MFCC feature extraction.

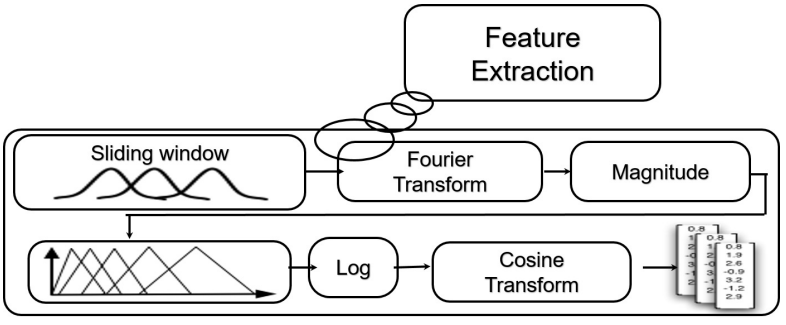


FIGURE 3.3: MFCC Feature Extraction [127]

One of the disadvantages of MFCCs is that they are not very robust against additive noise, and so it is common to normalize their values in speech recognition systems to lessen the influence of noise.

MFCCs are used for voice/speaker recognition. However, the indoor environmental audio signals had significant information at the trajectories of the MFCCs over

time. Therefore, we included the delta values and delta-delta values [128]. To compute these features, we used the *mfcc* function of the LibROSA library [129], which return 39 MFCC features per frame: 13 MFCCs where the zeroth coefficient was replaced with the logarithm of the total frame energy, 13 delta features and 13 delta-delta features.

3.2.3.2 Discrete Wavelet Transform

The DWT provides a compact representation of a signal in time and frequency and can be computed efficiently using a fast, pyramidal algorithm. In the pyramidal algorithm the input signal is analyzed at different frequency bands with different resolution by decomposing it into a coarse approximation and detail information. This is achieved by successive high pass and low pass filtering of the time domain signal. We used an 8-level DWT with the 20-coefficient wavelet family (db20) proposed by Daubechies [130], because of its robustness to noise, and extracted the mean and variance in each sub-band, resulting in 16 (high-frequency) features. The wavelet transform concentrated the signal features in a few large-magnitude wavelet coefficients; hence the coefficients with a small value (noise) could be removed without affecting the input signal quality.

In the kitchen environment signals, high frequency components are present very briefly at the onset of a sound while lower frequencies are present for a long period.

3.2.3.3 Zero-Crossing Rate

In the context of discrete-time signals, a zero crossing is said to occur if successive samples have different algebraic signs. The rate at which zero crossings occur is a simple measure of the frequency content of a signal.

The zero-crossing rate returned a 1×173 vector for each recording and we calculated the mean and median of each vector, resulting in two ZCR features per recording.

3.2.3.4 Spectral Roll-off

SR is defined as the frequency below which a certain percentage (85% - 95%; depending on the application) of the magnitude distribution of the power spectrum is accumulated. The equation of the feature is given in Equation (3.1):

$$\underset{m\in\{1,\ldots,N\}}{\operatorname{argmin}} \sum_{k=1}^{m} X_i(k) \geq C \sum_{k=1}^{N} X_i(k) \tag{3.1}$$

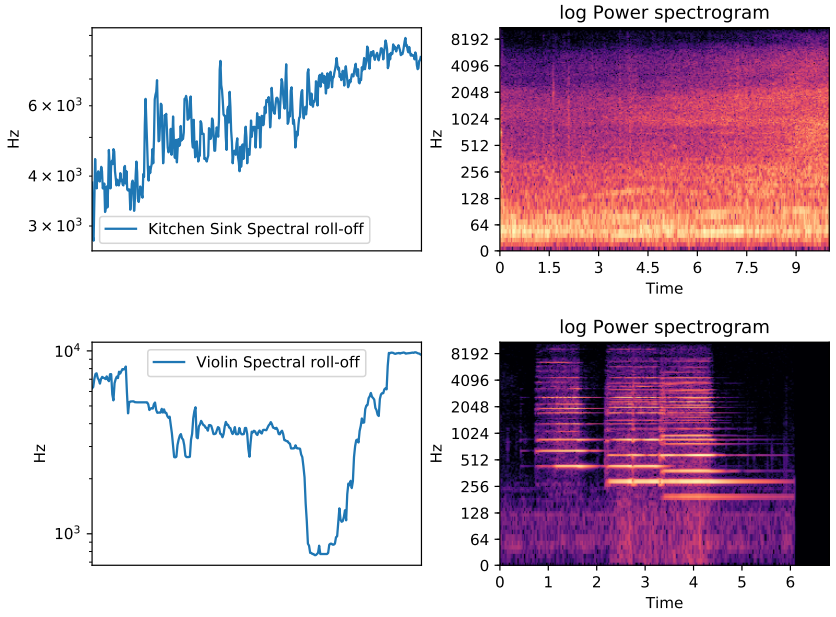


FIGURE 3.4: SR comparison between the sound of the kitchen sink (top) and the sound of a violin (bottom). The x-axis shows the time in s

where $X_i(k)$, $k = 1, ..., N$ are the Discrete Fourier Transform (DFT) coefficients of the i-th short-term frame and N is the number of frequency bins. The DFT coefficient $X_i(m)$ corresponds to the SR of the i-th frame, m is the roll-off frequency and C is the percentage of the magnitude distribution of the spectrum. We found a threshold of 95% to be suitable for distinguishing different kitchen sounds. Therefore C=0.95 in Equation 3.1. The mean and median of the SR for each recording were calculated and normalized between 0 and 1.

Figure 3.4 shows the difference of the SR between a violin recording and the running tap water in the sink. The harmonics of the violin are very distinct in the spectrum, the mean is 0.423 and the median is 0.417. On the other hand, the mean and median of the kitchen sink sound are 0.811 and 0.803 respectively.

3.2.3.5 Spectral Centroid

SC is defined as the "center of gravity" of the spectrum. It is described by Equation (3.2)

$$SC = \frac{\sum_{k=1}^{N}(k+1)X_i(k)}{\sum_{k=1}^{N}X_i(k)} \tag{3.2}$$

where $X_i(k)$, $k = 1, ..., N$ are the DFT coefficients of the i-th short-term frame and N is the number of frequency bins.

SC is directly related to the sharpness (high-frequency content) of the sound spectrum. Hence, higher SC values mean that there is a very bright sound with high frequencies present. The mean and median of the SC for each recording were calculated and normalized between 0 and 1.

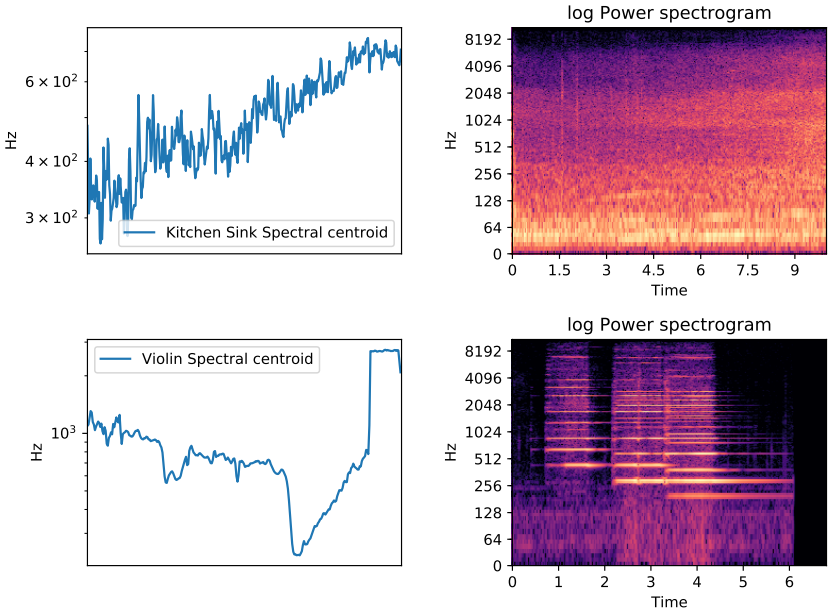


FIGURE 3.5: SC comparison between the sound of the kitchen sink (top) and the sound of a violin (bottom). The x-axis shows the time in s

Figure 3.5 shows a significant difference between the brighter sound of a violin and the more broadband sound of the running water of a kitchen sink. More specifically, for the kitchen sink, where low frequencies are mainly present, the mean is 0.126 and the median is 0.113. On the other hand, the "sharper" sound of the violin, where the harmonics are very distinct at higher frequencies has a mean of 0.383 and a median of 0.366.

3.2.3.6 Gammatone Frequency Cepstral Coefficients

The Gammatone filter-bank consists of a series of band-pass filters, which model the frequency selectivity property of the basilar membrane. The main difference

between the MFCC and GFCC is that the Gammatone filter-bank and the cube root are used before applying the DCT while the triangular filter-bank and the log operation are applied in MFCC. Equation (3.3) describes the calculation of the GFCC:

$$GFCC_m = \sqrt{\frac{2}{N}} \sum_{n=1}^{N} \log(E_n) \cos[\frac{\pi n}{N}(m - \frac{1}{2})], 1 \leq m \leq M \tag{3.3}$$

where E_n is the energy of the signal in the n-th band, N is the number of Gammatone filters and M is the number of GFCC.

We extracted 39 GFCC features per frame. These consisted of 13 GFCCs, 13 delta values and 13 delta-delta values.

3.2.4 Feature Selection

Feature selection was a crucial step for the first AED approach, since the framework had to detect activities in real-time.

3.2.4.1 Feature Aggregation

Out of the 5,985 recordings (original=1,995 and two augmented=3,990), we extracted the following features: 173×16 (DWT) + 2 (ZCR) + 2 (SR) + 2 (SC) + 173×39 (GFCC) + 173×39 (MFCC). Aggregating all the features into a single vector is an important step before passing it to the sequential backward search algorithms and applying principal component analysis. Feature extraction and classification (using the first AED approach) ran on a Raspberry Pi 3 Model B platform.

3.2.4.2 Sequential Backward Selection

SBS starts from the whole feature set $X = \{x_i \mid i = 1, \ldots, N\}$ and discards the "worst" feature (x') at each step, such that the reduced set $X - \{x'\}$ gives the maximum value of an objective function $J(X - \{x'\})$. Given a feature set, SBS gives better results but is computationally more complex than other statistical feature selection methods [131]. With SBS, we reduced the number of features to 17 per recording. The most important features were the DWT, the first five

GFCCs, first three MFCCs and the spectral centroid. The ZCR did not affect the overall system.

3.2.4.3 Principal Component Analysis

The central idea of PCA is to reduce the dimensionality of a dataset that consists of many interrelated variables, while retaining as much as possible the variation present in the dataset. We applied PCA to the features given by SBS to reduce the feature space down to two principal components. The principal components were used as input to the classifier.

3.2.5 Activity Classification

For the first AED approach, we compared the performance of a kNN classifier with 5 nearest neighbors, an SVM with a linear and a RBF kernel, an Extra Trees classifier, a Random Forest and a Gradient Boosting classifier.

For the second AED approach, we implemented a CNN based on a modified AlexNet [132] architecture. The CNN was trained on an NVIDIA GeForce GTX 1080 Ti.

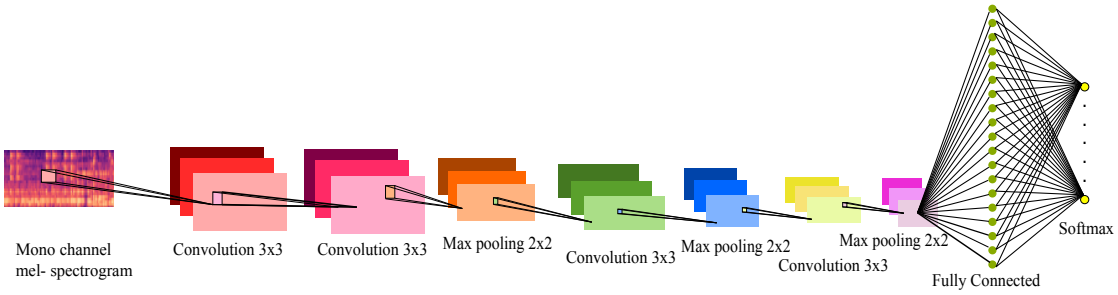


FIGURE 3.6: Second proposed AED approach

The CNN consists of 4 convolutional layers (Figure 3.6). The number of filters at each layer increases as a power of two. Specifically the first layer has 8 filters, the second 16, the third 32 and the fourth one 64. The first layer performs convolutions over the spectrogram of the input segment, using 3×3 kernels. The output is fed to a second convolutional layer which is identical to the first. A 2×2 max pooling operation follows the second layer and the subsampled feature maps are fed to two consecutive convolutional layers, each followed by max pooling operations. Each convolution operation is followed by batch normalization [133] of its outputs, before the element-wise application of the ELU activation function [134] to facilitate

training and improve convergence time. We selected the ELU activation function based on the results obtained by Clevert et al. [134], where it outperformed other commonly used activation functions (e.g., ReLU), when tested on image datasets using deep neural networks with more than five layers. After each max pooling operation, we apply dropout [135] with an input dropout rate of 0.2. The number of kernels in all convolutional layers is 5. The resulting feature maps of the consecutive convolution-max pooling operations are then fed as input to a fully-connected layer with 128 logistic sigmoid units to which we also apply dropout with a rate of 0.2, followed by the output layer which computes the softmax function. Classification is obtained through hard assignment of the normalized output of the softmax function

$$c = \underset{i=1,\ldots,N}{\operatorname{argmax}}\, y_i \tag{3.4}$$

$$y_i = \frac{\exp x_i}{\sum_{j=1}^{N} \exp x_j} \tag{3.5}$$

where N is the number of classes and x_i is the probability for the *i-th* class. We used the Adam optimizer [136] and trained our network with an initial learning rate l_r=0.001, which was reduced by a factor of 0.01 when there was no validation loss (categorical cross-entropy) improvement for five consecutive epochs. This ensured that there was no overfitting in the training. We trained the CNN for 20 epochs.

3.3 Results

In this section, we present experiments to assess the performance of our two AED systems. In all experiments, we used 80% of the dataset for training and 20% for testing. For all classifiers, the split between training set and testing set was identical, as is common in the literature [137]. In Section 3.3.1, we compare several classifiers for the first system, we select the one with the highest F1-score and recognition accuracy and compare the performance to that of the second system. In Section 3.3.2, we study the effect of feature fusion on the recognition rate of the best classifier identified in Section 3.3.1 (Gradient Boosting). In Section 3.3.3, we study the recognition accuracy as a function of signal duration. In Section 3.3.4, we analyze the effect of both the SNR and distance between the microphone

and event on the recognition accuracy in an "untrained" environment. In Section 3.3.5, we examine the response of the second AED system for an activity that was not included in the training set.

3.3.1 Comparison of the recognition accuracy of traditional classifiers against CNNs for AED

Table 3.2 compares the performance of various classifiers for the selected features. For all classifiers, we used the implementations in the scikit-learn [138] library. The signals used for this experiment were all the recordings from the three environments mentioned in Section 3.2.1 in addition to the Freesound recordings.

TABLE 3.2: Classifier Performance Comparison

MFCC+GFCC+SR+SC+ZCR+DWT (with augmented data)				
Classifier	**PRECISION**	**RECALL**	**F1-SCORE**	**ACCURACY**
kNN (5 nearest neighbors)	78.4%	79.4%	78.9%	79.4%
SVM (linear kernel)	79%	81.2%	80.1%	83.5%
SVM (RBF kernel)	84.1%	90.1%	87%	90.9%
Extra Trees	83.4%	85%	84.2%	89.7%
Random Forest	88.5%	89.1%	88.8%	91%
Gradient Boosting	***90.4%***	***90%***	***90.2%***	***91.7%***
Mel-Spectrogram (with augmented data)				
2D CNN /w 2D Max-pooling	***94.6%***	***90.9%***	***92.7%***	***96%***
1D CNN /w 1D Max-pooling	90%	89.7%	89.8%	91.3%

For the Random Forest classifier, we noticed, as the theory suggests, that increasing the number of trees can give a better and more stable performance; hence there is a small possibility of overfitting. The number of leaves in the tree had to be small, in order to capture noisy instances in the training dataset [139]. Therefore, we selected 50 samples for each leaf node, after performing a grid search at 25, 50, 75 and 100. For the RBF-based SVM classifier, the highest values for all evaluation measures were found for $\sigma = 1$ and $C = 0.1$. The parameter σ of the RBF kernel handles the non-linear classification and parameter C trades off correct classification of training examples against maximization of the decision function's margin. Finally, for Gradient Boosting 500 estimators were picked. The deviance (logistic regression) loss for classification with probabilistic outputs, was used, since it was a multi-class problem. Another important parameter that affected the classification performance was the learning rate. All values from 0.01 to 0.1 with a 0.01 step, were tried and 0.05 was selected, as it provided the best results. We kept the rest of the parameters in the scikit-learn library at default settings. Additionally, as Gradient Boosting is fairly robust to overfitting, the large number of estimators resulted in a better performance, achieving an F1-Score of 90.2%. Good results

for boiling, frying, the use of the mixer, and also the use of the dishwasher, were obtained. However, the activity of the "running" kitchen faucet was "understood" by the architecture as doing the dishes because some recordings were very similar due to the timing (meaning that no dishes or utensils were "heard" from the microphone).

We also applied McNemar's test to determine whether there was a significant difference between the accuracy of the classifiers. The results are summarized in Table 3.3 and show in particular that the 2D CNN classifier is statistically different from all other classifiers at the 0.05 significance level.

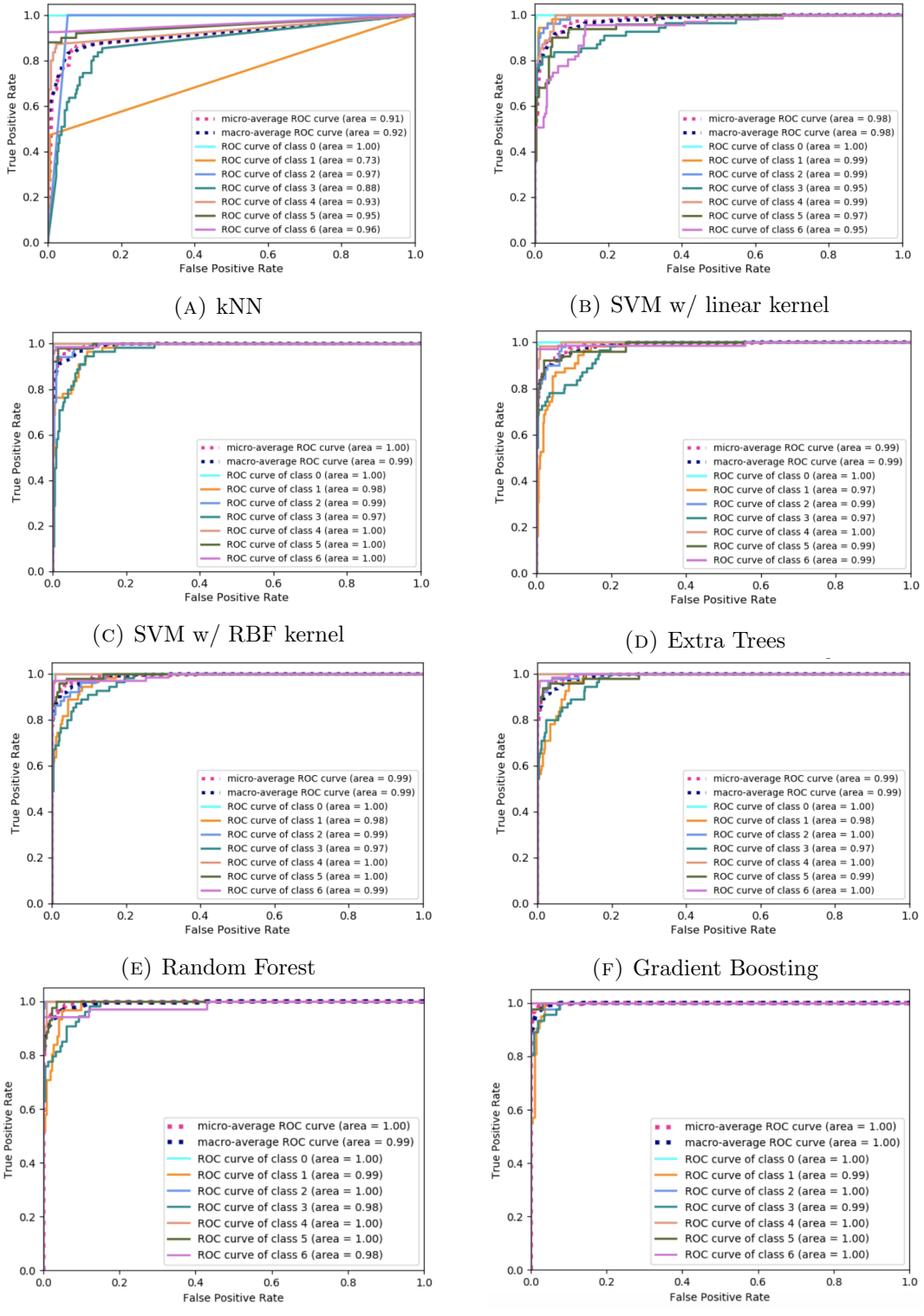
To further compare the performance of the classifiers, we plotted their ROC curves (Figure 3.7). We noticed that the boiling class was the most easily separable class for all the classifiers. The classes of cutting the bread and operating the kitchen faucet were the hardest ones for all the classifiers. This is because many recordings had sounds corresponding to these two particular classes towards the last second of the 5 s-recording.

In the following experiments, the first AED system was used with the Gradient Boosting classifier, since it achieved the highest performance characterized by a stable relationship between precision, recall, F1-Score and recognition accuracy.

TABLE 3.3: McNemar's test results

p-values (statistically significant where p <0.05)								
Classifiers	kNN	SVM Linear	SVM RBF	Extra Trees	Random Forest	Gradient Boosting	2D CNN	1D CNN
kNN	-	0.01337	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001
SVM Linear	-	-	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001
SVM RBF	-	-	-	0.52239	0.86793	0.51137	<0.001	0.74282
Extra Trees	-	-	-	-	0.24778	0.05247	<0.001	0.21532
Random Forest	-	-	-	-	-	0.62905	<0.001	1
Gradient Boosting	-	-	-	-	-	-	0.00259	0.82380
2D CNN	-	-	-	-	-	-	-	<0.001

In order to highlight the importance of 2D max-pooling, we compared it to 1D max-pooling with a 1D CNN. The input to the 1D CNN network were mel-spectrograms with 128 bins. The resulting feature matrix input vector to the 1D CNN consisted of 128 mel-band energies in 431 successive frames (number of FFT samples = 1024 with hop length = 512, or window size of 20 ms with a 10 ms hop size from the



(A) kNN (B) SVM w/ linear kernel

(C) SVM w/ RBF kernel (D) Extra Trees

(E) Random Forest (F) Gradient Boosting

(G) 1D CNN (H) 2D CNN

FIGURE 3.7: ROC curves for the selected classifiers. Classes 0, 1, 2, 3, 4, 5 and 6 correspond to boiling, cutting bread, dishwasher, doing the dishes, frying, operating the kitchen faucet and mixer, respectively

next one). The 1D CNN had the same number of filters, kernels, etc. as the 2D CNN (described in Section 3.2.5). The main differences between the two networks are that kernels change from 3×3 to 3×1, max-pooling from 2×2 to 2×1 and in the Keras [140] library the $Conv2D$ and $MaxPooling2D$ layers are replaced with the $Conv1D$ and $MaxPooling1D$, respectively. The 2D CNN with 2D max-pooling, was able to capture the spatio-temporal information of the given signal and achieved an F1-Score of 92.7%. On the other hand, the 1D CNN achieved an F1-Score of 89.8% only. This showed that the audio signals that were present in the kitchen environment contained important information in the frequency domain.

3.3.2 Fusion of features for the first AED approach

Figure 3.8 shows how fusing features improves the performance of the first AED approach with the Gradient Boosting classifier. The accuracy rates were calculated for seven feature combinations.

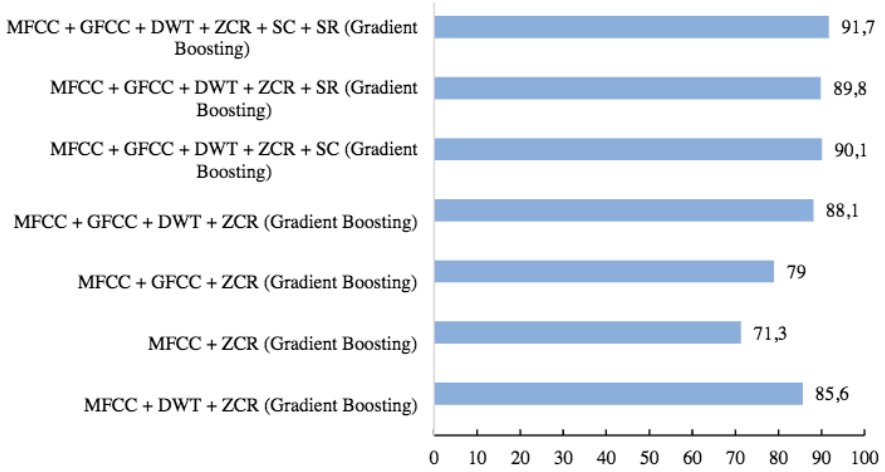


FIGURE 3.8: Recognition accuracy for different audio features using Gradient Boosting. The results are after the aforementioned feature selection.

Many sounds in a kitchen environment have an interchangeable pattern (bigger/smaller values for odd/even MFCCs). Some mechanical noises (mixer, dishwasher) have high short-time energy on their fundamental frequency and others (forks, spoons, trays) have high short-time energy on higher frequencies. This served as our motivation to test more time-frequency features in the kitchen recordings. Specifically, when introducing the GFCCs and the DWT, the recognition accuracy was significantly improved. MFCCs and ZCR achieved an accuracy of 71.3%. When we added the GFCCs first and DWT second, the accuracy improved to 79% and 85.6% respectively. GFCCs use the ERB scale. The ERB scale has a finer

TABLE 3.4: McNemar's test on the features

p-values (statistically significant where p <0.05)							
Features	MFCCs + GFCCs + DWT + ZCR + SC + SR	MFCCs + GFCCs + DWT + ZCR + SR	MFCCs + GFCCs + DWT + ZCR + SC	MFCCs + GFCCs + DWT + ZCR	MFCCs + GFCCs + ZCR	MFCCs + ZCR	MFCCs + DWT + ZCR
MFCCs + GFCCs + DWT + ZCR + SC + SR	-	0.23788	0.14346	0.01612	<0.001	<0.001	<0.001
MFCCs + GFCCs + DWT + ZCR + SR	-	-	1	0.24778	<0.001	<0.001	<0.01609
MFCCs + GFCCs + DWT + ZCR + SC	-	-	-	0.16863	<0.001	<0.001	0.00642
MFCCs + GFCCs + DWT + ZCR	-	-	-	-	<0.001	<0.001	0.15385
MFCCs + GFCCs + ZCR	-	-	-	-	-	0.00239	0.00254
MFCCs + ZCR	-	-	-	-	-	-	<0.001

resolution at low frequencies, which were present in the kitchen environment, compared to the mel-scale used by the MFCCs. Additionally, the DWT was able to separate the fine details of the input signal and increased the recognition accuracy. As for the classifiers, we applied McNemar's test on the features (Table 3.4) to check the statistical significance of the results.

3.3.3 Recognition accuracy as a function of the audio sample duration

We studied the impact of segment duration on the accuracy of activity recognition within the kitchen environment. Figure 3.9 shows that a 3 s time duration of the input signal is sufficient for accurate activity recognition. For the Gradient Boosting classifier, we noticed an unexpected drop-off for the activity of doing the dishes after three seconds. Examination of the confusion matrices revealed that there is a recognition uncertainty of the activity of doing the dishes and the operation of the kitchen sink. After careful listening of all the recordings, we noticed that there were times when the faucet was turned on and only at the last second of the recording an object (plate, utensils) was picked to be washed. On

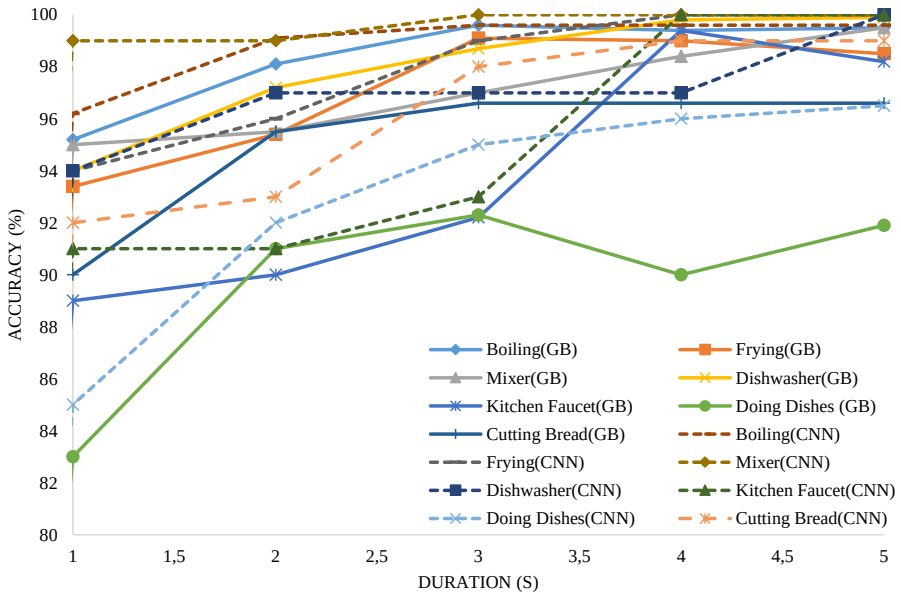


FIGURE 3.9: Recognition accuracy (using the Gradient Boosting classifier and the CNN) as a function of the sample duration

the other hand, the performance of the CNN improved as the audio clips became longer, since it was able to find clear patterns in the mel-spectrogram image.

3.3.4 Dependence of recognition accuracy on certain distance and SNR in a new environment

We trained both systems in the environments of Figure 3.2 (a-b) and tested them in the environment of Figure 3.2 (c). Our training set consisted of 1547 recordings for the seven classes. We tested the systems on the following classes only: *dishwasher, mixer, utensils/trays, kitchen faucet*. For this experiments, we renamed the class *doing the dishes* to *utensils/trays*, since the people in the kitchen rinsed the utensils/trays for a very short period of time and then used the dishwasher. 41 recordings for the activity of moving the utensils/trays were collected in order to test the two AED systems.

TABLE 3.5: Confusion matrix using Gradient Boosting for the classes of the framework in a new environment (not included in the training dataset). The distance between the microphone and each activity was 3 m

	Mixer	Dishwasher	Utensils/Trays	Kitchen Faucet
Mixer	45	11	0	1
Dishwasher	0	48	2	10
Utensils/Trays	0	1	39	0
Kitchen Faucet	0	5	0	40

TABLE 3.6: Confusion matrix using CNN for the classes of the framework in a new environment (not included in the training dataset). The distance between the microphone and each activity was 3 m

	Mixer	Dishwasher	Utensils/Trays	Kitchen Faucet
Mixer	45	11	0	0
Dishwasher	0	46	0	5
Utensils/Trays	0	2	41	0
Kitchen Faucet	0	6	0	46

For this experiment we could not test all seven classes since, i) the recordings from cutting the bread were collected and trained using the setup of that environment, ii) there was no frying activity due to dietary instructions from the elderly care home where the experiment took place and iii) the setup was similar to a restaurant kitchen setup and we could not detect the boiling activity (the microphone was placed at a large distance from the stove). The results (Table 3.5 and Table 3.6) show that even with a relatively small training dataset and a distance of 3 m from the event to be classified, we were able to obtain satisfactory results when testing in a new indoor environment. For the Gradient Boosting system, the overall weighted recognition accuracy was 92%, whereas the 2D CNN achieved a recognition accuracy of 93.88%.

TABLE 3.7: Recognition accuracy of Gradient Boosting and CNN according to distances and SNRs

Activities	**Distance (m)**	**SNR (dB)**	**Accuracy with Gradient Boosting (%)**	**Accuracy with CNN (%)**
Using the Kitchen Sink	3	-27	90.2	*93.4*
	0.4	-10	94	*98.8*
Using the Mixer	6	-11	*98.5*	97.1
	3	-8	100	100
Moving the Utensils/Trays	6	-16	91.1	*95*
	3	-13	96.8	*100*
Using the Dishwasher	3	-30	*90.2*	89.9
	1	-25	*93*	91.7

The distance between the activity and the microphone affected the recognition accuracy. Table 3.7 shows the SNR and the classification accuracy, using the Gradient Boosting and the CNN, of a set of activities at various distances. The ambient noise of the kitchen at AKTIOS (fan and refrigerator at -32 dB) operating at the time of the experiment dropped the performance of the approaches when increasing the distance from the microphone. The CNN outperformed the Gradient Boosting except when the mixer was used and the microphone was placed 6 m

and 3 m away or when the dishwasher was used and the microphone was placed 1 m away.

3.3.5 Tests with activity that was not included in the training set (coffee machine) using the second AED system

For this experiment, we used the MEMS microphone board and a laptop 50 cm away from a coffee machine to collect 25 recordings of 5 s each. Out of the 25 recordings, 8 were classified as boiling, since it was the closest match in terms of the audio characteristics of the filter coffee machine. For the remaining 17, the classifier output was discarded because the output probability for each class was below the minimum threshold, set for this experiment to 0.7. More precisely, the class probability was between 0.5 and 0.6 for the boiling class and randomly distributed among the other classes.

3.4 Conclusions

We proposed two systems for AED in real-world conditions. The first one relies on feature extraction, selection, and classification, while the second one uses a CNN to learn from mel spectrogram images without the need for human-crafted features. Adding more audio features does not necessarily increase the recognition accuracy of the first system. However, feature selection methods and feature dimensionality reduction techniques, are critical to the success of the system. GFCCs and DWT coefficients significantly increased the recognition accuracy. They outperformed other well-known time-frequency features in the presence of background noise. Furthermore, we found that a signal duration of 3 s provided a good trade-off between time delay and recognition accuracy. The two systems were tested in a new environment and provided recognition accuracies above 90% for appliances that were up to 6 m away. This is a positive result since in most commercial kitchen environments, the distance between the microphone and the target appliance will be smaller.

Finally, in order to check the robustness of our second AED system, we tested it on an activity that was not included in the training set. The system correctly

rejected the recording in 68% of the cases and misclassified it as boiling in the remaining cases.

Chapter 4

Convolutional Recurrent Neural Networks for Speech Activity Detection

4.1 Introduction

One of the most important problems in the area of speech signal processing is distinguishing speech from non-speech periods in an audio signal [141]. SAD is part of many applications (e.g., ASR [142] and speaker diarization [143]). Recently, SAD has received attention especially in research projects [144] and challenges [145, 146]. The main reason is that speech recordings and specifically historical recordings, such as the Apollo audio data [147], are characterized by multiple noise types and several overlap instances over most channels. Most audio channels are degraded due to high channel noise, system noise, attenuated signal bandwidth, transmission noise, cosmic noise, analog tape static noise, and noise due to tape aging.

SAD algorithms have been extensively researched [148, 149, 150, 151]. These algorithms are mainly probabilistic models, use temporal and power spectral characteristics of sound and do not require training. Because of their low complexity, the majority of SAD algorithms work well for real-time applications. However, they require extensive fine-tuning of their hyper-parameters and have lower performance, to a certain extent, in low SNR environments.

A large number of SAD methods and models have been proposed for highly degraded acoustic conditions. Most of them are supervised and exploit the time and frequency properties of speech and noise to effectively separate speech from non-speech [152, 153]. Some of them use energy operators and multi-band modulations [154], autocorrelation coefficients [155], as well as time and frequency feature-level fusion [156].

The problem with most supervised methods is that the data has to be well annotated (in milliseconds), which is a time-consuming task requiring specialists to hand-label the audio data. To address this problem, semi-supervised and unsupervised methods have been proposed. In particular, GMMs have been extensively used [157, 158]. GMM-based SAD systems are typically composed of two GMMs: one trained on speech frames and one on non-speech frames.

More recently, DL based methods have been proposed to solve the SAD problem. The ability of deep neural networks to automatically extract low-level features from a given signal segment, has made them popular in various scientific fields. Various DL methods have been explored, compared and fused with SAD algorithms [159], in order to select the ones best suited for the SAD problem [160]. Among these DL based methods, RNN have several properties that make them a popular choice for SAD [161, 162].

In this Chapter, we consider SAD as a multilabel classification problem, meaning that each sample in an audio recording has a unique label (e.g., 8000 samples equal to 8000 labels in a 1 s recording sampled at 8 kHz) and use a 2D CRNN to address it. The ability of the CNN layers to capture the temporal and frequential information of the audio signal and the ability of the recurrent layers to identify the time intervals, for long sequences, of the classified events (speech, non-speech), make them suitable for this problem. Furthermore, we use the stratified k-fold cross-validation method, to preserve the percentage of samples for each class in different folds and use majority voting to calculate the DCF. Finally, we perform convolutions on the k-fold majority voting results to smooth the output. The main contribution is related not only to the simplicity of the proposed frameworks (e.g., end-to-end CRNN based on raw waveforms), but the importance of the post-processing step is highlighted. Performing convolutions with ones for every 10 ms can successfully clear all the false positive spikes (speech detection) of the network prediction.

The remainder of the Chapter is organized as follows. Section 4.2 describes our methodology, including raw audio signal pre-processing, feature extraction and

network architecture. The evaluation of the networks for the dataset is presented in Section 4.3 and the results in Section 4.4. Finally, Section 4.5 concludes the Chapter [163].

4.2 Proposed Approach for Speech Activity Detection

This Section describes the steps of our proposed approach, starting from the extraction of the features of the audio signal that are used as input to the 2D network architectures, to describing the neural network architectures used in our experiments. As an augmentation method, we used random time shifts for each 1 s of recording (-8000 samples, +8000 samples), creating an array with elements that roll between the last position and the re-introduced at first in order to keep all the signal information. Finally, we did not apply any denoising method, since we wanted the network to learn how to distinguish between noisy and ambient recordings.

4.2.1 Feature Extraction

As a first step, we split the recordings into segments of 1 s and assign to each of the samples the corresponding label (0: non-speech, 1: speech). The main advantage of deep neural networks is their ability to extract features from raw data. We calculated the STFT spectrogram for each 1 s - recording and extracted the corresponding grayscale spectrogram image. The length of the FFT was 256, with a hop length of 64. We selected the Hanning window for the FFT, since it is commonly used for speech signals [164]. This resulted in a 129x126 spectrogram used as input to the 2D CRNN.

4.2.2 Convolutional Recurrent Neural Networks Description

As a network architecture, we used a modified 2D CRNN, where we permute the dimensions of the CNN output and then reshape them to feed the GRU of the RNN. Additionally, we apply max-pooling on the frequency domain only, when

calculating convolutions, allowing the entire time information to be processed by the RNN. The network was trained in Keras [140] with TensorFlow [165] backend, using a batch size of 32 for 25 epochs.

Our 2D CRNN architecture is shown in Figure 4.1. The architecture was inspired by Bartz *et al.* [166], who applied 2D CRNNs for language identification in text documents. We applied a similar architecture for SAD. The CNN part of our 2D CRNN architecture consists of five convolutional layers. The first one has 16 filters,

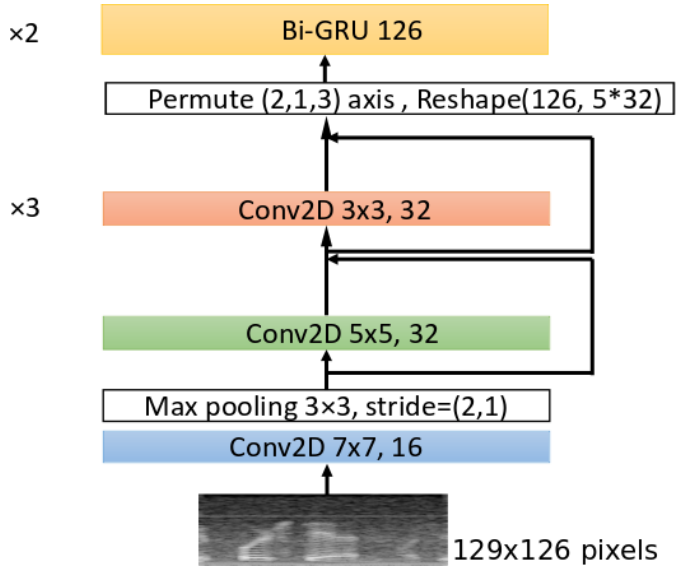


FIGURE 4.1: 2D CRNN architecture with STFT spectrogram magnitude representation as input. The arrows around the Conv2D blocks indicate the same max-pooling operation applied after each convolutional layer

the second 32, the third 32, the fourth 32 and the fifth one 32. The first layer computes convolutions over the time and frequency domain, using 7×7 kernels. A 3×3 max pooling operation follows each convolutional layer and the subsampled feature maps are fed to the next convolutional layer. We used 2×1 strides for each max pooling operation since we wanted to sub-sample the frequency domain and leave the time domain as is to be processed by the RNN part. The size of the kernels was decreased to 5×5 in the second convolution and to 3×3 in the third, fourth and fifth. Each convolution was followed by batch normalization [133] of its outputs, before the element-wise application of the rectified linear unit (ReLU) [167] activation function. Finally, the resulting feature maps of the consecutive convolution-max pooling operations were permuted and reshaped to be used as input to two bi-directional GRUs (RNN part), each one having a filter size of 126. We used the Adam [136] optimizer with an initial learning rate l_r=0.001 which was reduced by a factor of 0.01, when there was no DCF improvement for 5 consecutive epochs.

4.3 Evaluation and Analysis

We conducted experiments on the development and evaluation datasets of the Fearless Steps Challenge (Apollo 11) [168]. These datasets consist of 39 and 40 recordings, respectively, each recording containing a total of approximately 30 min audio in wav format and sampled at 8000 Hz.

4.3.1 Network Architectures

The proposed 2D CRNN model with STFT spectrograms as input was compared to a 1D CRNN that uses raw waveforms as inputs, the 2D CRNN where the STFT spectrograms were replaced by MFCC images, a GRU-RNN [161] using MFCCs as input, a state-of-the-art VAD system [169] and the baseline system results [147], provided by the organizers. MFCCs are one of the most popular features for voice recognition [126]. For our experiments, we calculated 20 double-delta coefficients (including the $0th$ energy coefficient) using an FFT with a Hanning window size of 2048 and a hop length of 512, which resulted in a 20×16 vector. This 2D vector was used as an input to the 2D CRNN. The main reason for selecting the double-delta coefficients is that they convey richer information about the frame context [170].

The GRU-RNN is a basic network consisting of two bi-directional GRU units, each one having a filter size of 126. All the networks were trained using the same parameters as described in Section 4.2.2.

The 1D CRNN architecture is shown in Figure 4.2. The CNN part of it consists of five convolutional layers. The filter size at each layer increases as a power of two. Specifically the first one has 16, the second 32, the third 64, the fourth 128 and the fifth one 256. The first layer performs convolutions over the time domain (raw waveform), using 1×3 kernels. A 1×2 max pooling operation follows on each convolutional layer and the subsampled feature maps are fed to the next convolutional layer. Each convolution is followed by batch normalization of its outputs, before an element-wise application of the ReLU activation function. We selected this activation function for each layer, as it is the most commonly used. Finally, the resulting feature maps of the consecutive convolution-max pooling operations are fed as input to two bi-directional GRUs (RNN part), each one having a filter size of 126. The main reason for using bi-directional GRUs over the original LSTM units proposed for CRNNs, is that they train faster and we can

achieve comparable performance with the LSTMs. Furthermore the bi-directional unit, compared to a uni-directional unit [171], could learn the context based on future and past values (e.g., speech followed by non-speech, large periods of silence or noise). We used the Adam optimizer with an initial learning rate lr=0.001. We reduced the lr by a factor of 0.01, when there was no DCF improvement for five consecutive epochs, which boosted our DCF score. The 1D and 2D CRNNs use the same number of convolutional layers, but with different kernel sizes and number of filters, since the nature of the input data is different.

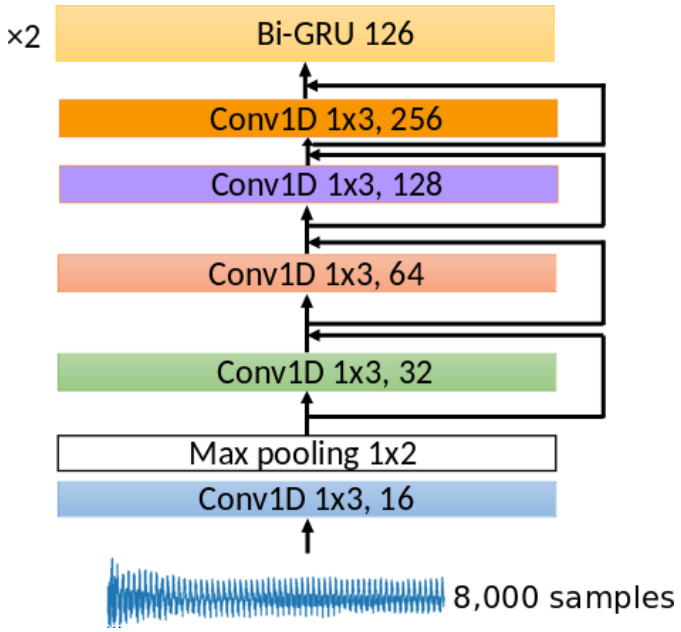


FIGURE 4.2: 1D CRNN architecture with raw waveform as input. The arrows around the Conv1D blocks indicate the same max-pooling operation applied after each convolutional layer

4.3.2 Network Training

In order to avoid bias over a single class, we trained our networks using the stratified k-fold method ($k = 5$). This helped us preserve the percentage of samples for each class. The metric that was selected as a performance measurement was the DCF score, which is defined as follows:

$$DCF(\theta) = 0.75 \times P_{FN}(\theta) + 0.25 \times P_{FP}(\theta)$$

where θ denotes a given system decision-threshold setting. P_{FP} is the probability of a false positive (FP), which is equal to the *total FP time* divided by the *annotated total non-speech time* and P_{FN} is the probability of a false negative (FN), which is

equal to the *total FN time* divided by the *annotated total speech time*. To optimize parameter θ we tested all values from 0 to 1 with a step size of 0.01.

4.4 Results

The results are summarized in Table 4.1. Our approaches significantly outperformed the unsupervised baseline algorithm and the VAD system [169] for the development set (ground truth given). VAD algorithms can predict continuous speech segments in some noisy channels but they require a lot of fine-tuning based on the recorded channel. Although the 2D CRNN achieved the best performance for the challenge, the 1D CRNN achieved a good DCF score. Considering that the 1D CRNN has a smaller number of trainable parameters, compared to the 2D CRNN, and uses raw waveforms as input, makes it ideal for real-life applications, since it is an end-to-end system.

TABLE 4.1: Performance of different architectures using DCF as a metric on the development dataset. No temporal collars are used

Systems	**DCF (%) without filtering**
1D CRNN	3.02
2D CRNN (STFT spec. image)	**2.89**
2D CRNN (MFCC image)	4.02
MFCC RNN	4.08
Google VAD [169] (mode 0)	13.99
Baseline [147]	8.6

Figure 4.3 depicts the SAD performance for the different architectures. We notice that the 2D CRNN with STFT spectrograms as input is able not only to accurately detect the speech and non-speech segments, but also to correct the labeling of the ground truth.

Since we evaluated the CRNNs using a 5-fold stratified cross validation, it was also necessary to compare the performance of each fold. Figure 4.4 shows that the average of the 5-folds achieved the best performance amongst them, and the standard deviation of each fold was very small, justifying the robustness of the CRNN architectures.

The most important advantage of the proposed method is a post-processing moving average filter, applied to the output of the CRNN. An array of ones was used

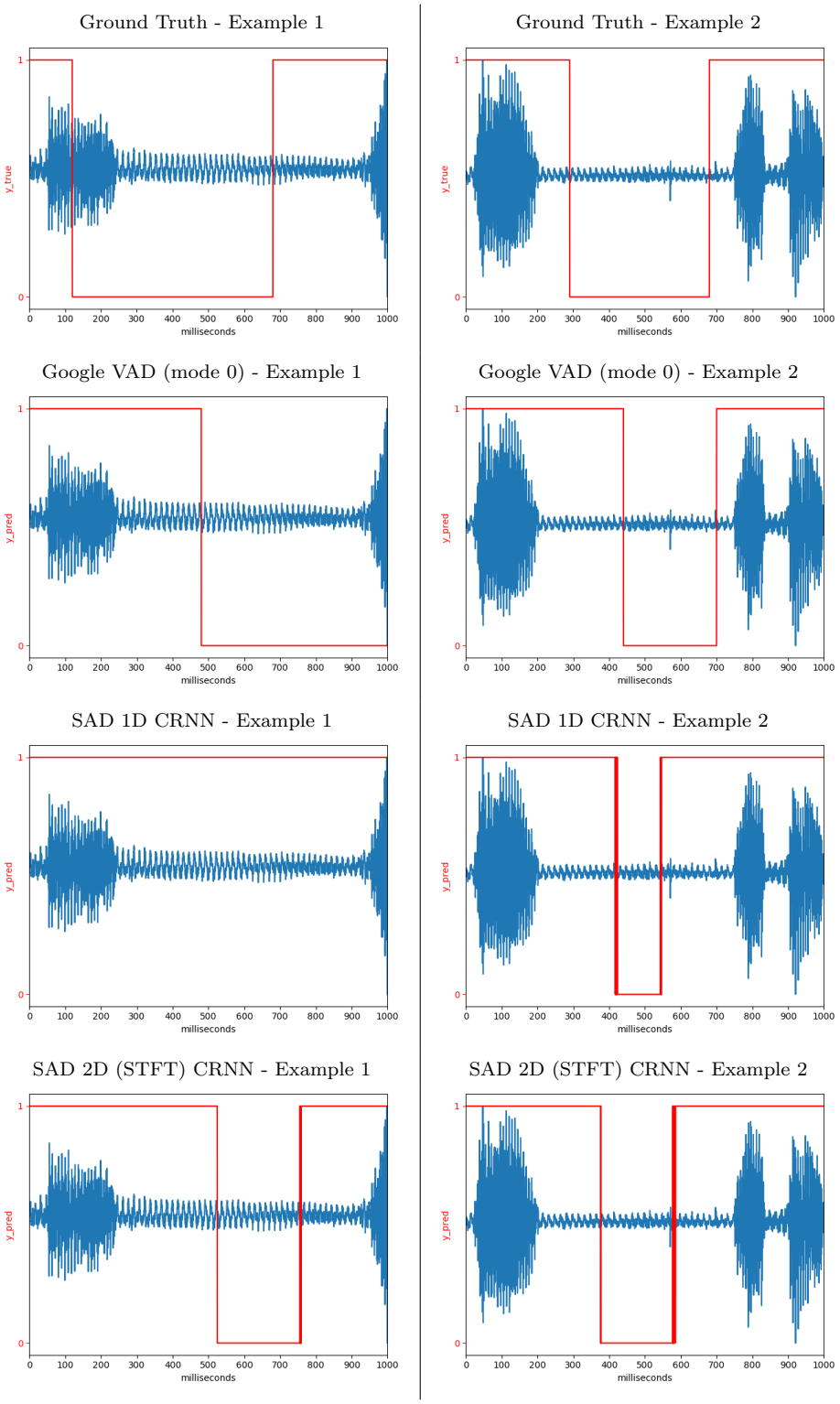


FIGURE 4.3: Examples of speech and non-speech activity detection of 1D and 2D (STFT) CRNN architectures with the ground truth of the development dataset

to perform convolutions, acting as a high-pass filter. By calculating convolutions of 10 ms windows (80 samples) average, the predictions (red line) of the CRNN

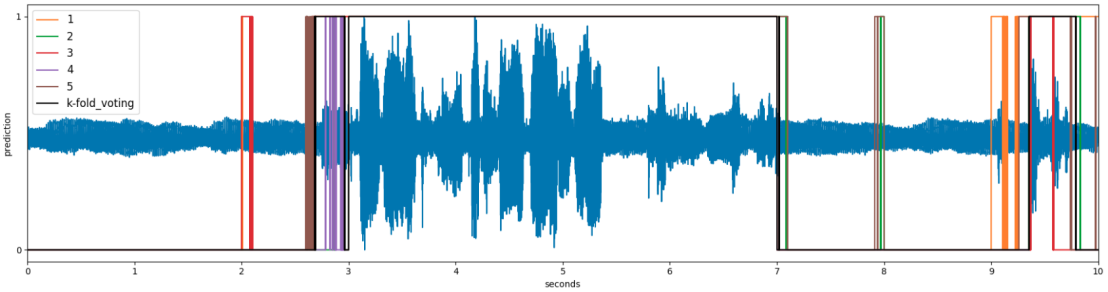


FIGURE 4.4: SAD results for the 5-folds and the ensembled majority on an example of the evaluation dataset (no ground truth given) using 2D (STFT) CRNN

were corrected (black line). Figure 4.5 shows the advantage of our moving average (temporal smoothing) post-processing filter. The CRNN architectures output many spikes in the waveform as speech predictions. These spikes usually range from 0.01 to 0.5 s (red line). Additionally, the average filter can also work as a confidence score for each predicted segment. The main problem that we are trying to solve is the misclassification of spikes (either detected as speech or non-speech). As another post-processing step, segments whose duration was shorter than 150 ms (1200 samples) and which were predicted as speech were relabelled as non-speech if their preceding and following segment was predicted as non-speech. This is because speech segments cannot be too short due to the inertia of human articulators.

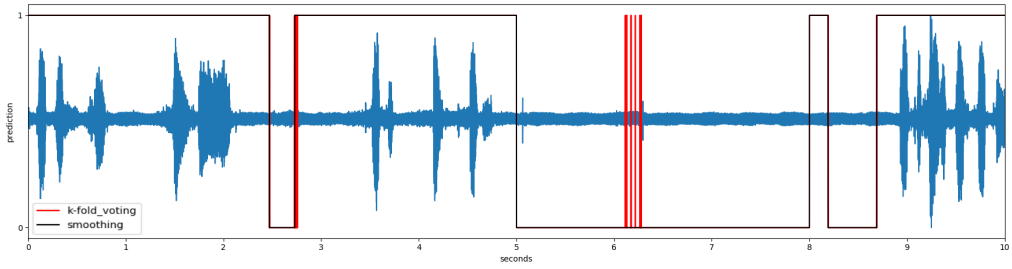


FIGURE 4.5: SAD results using the moving average filter of the convolutions (temporal smoothing) after averaging the 5 folds

Finally, our 2D CRNN architecture, using STFT spectrograms as input, achieved a DCF score of 3.318% on the evaluation dataset of the 2019 Fearless Steps SAD Challenge [147], ranking first among the 27 submissions.

4.5 Conclusions

We proposed a system that exploits a 2D CRNN for SAD. On Task 1 of the 2019 Fearless Steps Challenge, our system outperformed a well-known VAD algorithm [169] and achieved the first place among the 27 submissions. The novelty of our approach lies in treating SAD as a 2D image multilabel classification problem, where the input is an STFT spectrogram of the audio recording. The operational simplicity of our system makes it able to run on low-power devices and has been tested on a Raspberry Pi 3B+ model at CERTH's nZEB smart home.

Chapter 5

Energy-based Event Detection

5.1 Introduction

Knowing the true activity of occupants in a building at any given time is fundamental for the effective management of various building operation functions ranging from energy savings to security targets, especially in complex buildings with different internal kind of use. As the activities of occupants within the building vary throughout the day, it is difficult to characterize the different activities in different time periods. In general, activity monitoring in buildings is of high interest, since it significantly contributes to the improvement of a building's energy efficiency [3] and increases the quality of life of people in AAL environments [4]. Therefore, there is a need for detailed activity knowledge.

In this Chapter, we propose a decision engine that is able to identify the activities based on the energy consumption rate of household appliances using smart plugs to support NILM and machine learning. The human activity is recognized using only the energy consumption rate information from several appliances in a domestic environment by using only smart plugs. The decision engine for human activity recognition applies popular machine learning classifiers (supervised) on household appliances aiming to infer the appliance status (ON/OFF) along with a real time appliance activity proportion measurement for each appliance to determine daily household activities related to these appliances. To determine the most effective classifier for each appliance we run a series of Monte Carlo simulations testing different settings for each classification method.

Our work proposes the use of unobtrusive and easy-install tools (smart plugs) for data collection and a decision engine that combines energy signal classification using dominant classifiers (compared in advanced with grid search) and a probabilistic measure for appliance usage. It helps preserving the privacy of the resident, since all the activities are stored in a local database.

The remainder of the Chapter is organized as follows. In Section 5.2, we describe the energy consumption rate dataset and how we pre-process it. In Section 5.3, we formulate the decision engine for the activity recognition. In Section 5.4, we describe our simulation setup and give our results. In Section 5.5, we draw our conclusions.

5.2 Data Collection and Analysis

In order to infer the daily activities of a resident from electricity meters, one has to know the operating state of an electrical appliance. Estimating the operating state of an electrical appliance within a household, based on its power consumption, requires an extensive data collection procedure. As an initial step of this research, we focused on the power consumption of electrical appliances in a kitchen environment. From the first house (House A) (Figure 5.1a), we collected data from the oven, the cooker hood, the dishwasher, the fridge and the main consumption (which includes the Heating, Ventilation, and Air Conditioning, lights and other appliances) of the entire apartment. Regarding the second house (House B) (Figure 5.1b), we collected data from one fridge, since our goal in that specific setup, was to check if it is possible to detect when a resident opens and closes the fridge door. In what follows, the infrastructure for data collection and the approach that was followed for pre-processing of the dataset are described.

5.2.1 Data Collection Infrastructure

Figure 5.2 shows our data collection infrastructure. We installed a Gavazzi smart electricity meter in the oven and the main consumption panel of House A. The Gavazzi meter communicates with a Raspberry Pi via BACnet and then the Raspberry Pi sends the raw data to the InfluxDB database via a RESTful web service. We also measured the electricity consumption of selected devices (fridge, cooker hood, dishwasher and oven) via a wireless network of smart plugs that use the

(A) Kitchen environment setup CERTH nZEB smart home with selected devices of interest

(B) Kitchen environment setup CERTH/ITI ground floor kitchen with selected devices of interest

FIGURE 5.1: Data collection environments

ZigBee protocol (https://www.plugwise.com). The installed smart plug modules communicate with each other forming a network of mesh topology. Furthermore, a special built-in module was used in order to monitor the power consumption of the electrical kitchen appliance.

An aggregator application was developed and installed on a PC (MQTT Broker). It requested the current power consumption from each module for given time steps, received the corresponding messages, which include the measured energy consumption rate of the connected appliance in Watts, the time stamp (in UTC; later converted to local time), the ID of the device, and then stored the data directly into the database (InfluxDB).

For the second house (House B), we followed a similar procedure using the plugwise smart plugs that collected the energy consumption rate data for the specific device

that we monitored (fridge).

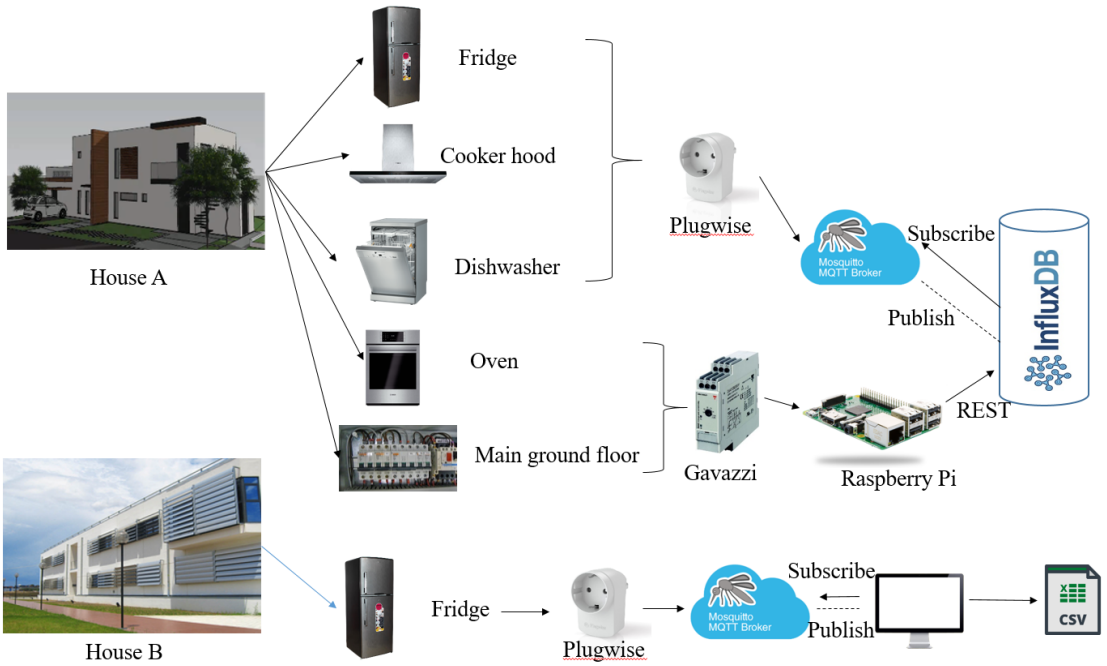


FIGURE 5.2: Data collection infrastructure

5.2.2 Data Pre-Processing

After retrieving the raw data for House A, over a period of one month, a pre-processing step was performed in order to create the final aggregated dataset, which includes events per one-minute intervals of all the measured features. It is worth mentioning that due to technical issues with the smart plugs or the InfluxDB database, we had to overcome the sparsity of the raw data matrix. In order to solve this problem, we filled the missing values with the mode of the values of the last 15 minutes, until a new value was sent to the database. Regarding House B, we collected the electricity consumption of a fridge over a period of 10 days. The plugwise smart plug was sending data every 5 seconds, a time interval that was sufficient to detect whether someone opens and closes the door of the fridge.

The next step was to aggregate the features, consisting of energy consumption rate in Watts for each of the four appliances of interest (oven, fridge, dishwasher, cooker hood). Firstly, we had to round the time (index), since there was a delay of a few ms between the "subscription" and the "publish" of the event to the MQTT broker. Secondly, we manually labeled the dataset regarding the "target feature" or the state of operation (ON/OFF). The fridge was considered to be always "ON", even when the compressor was not operating. The rest of the devices were labeled

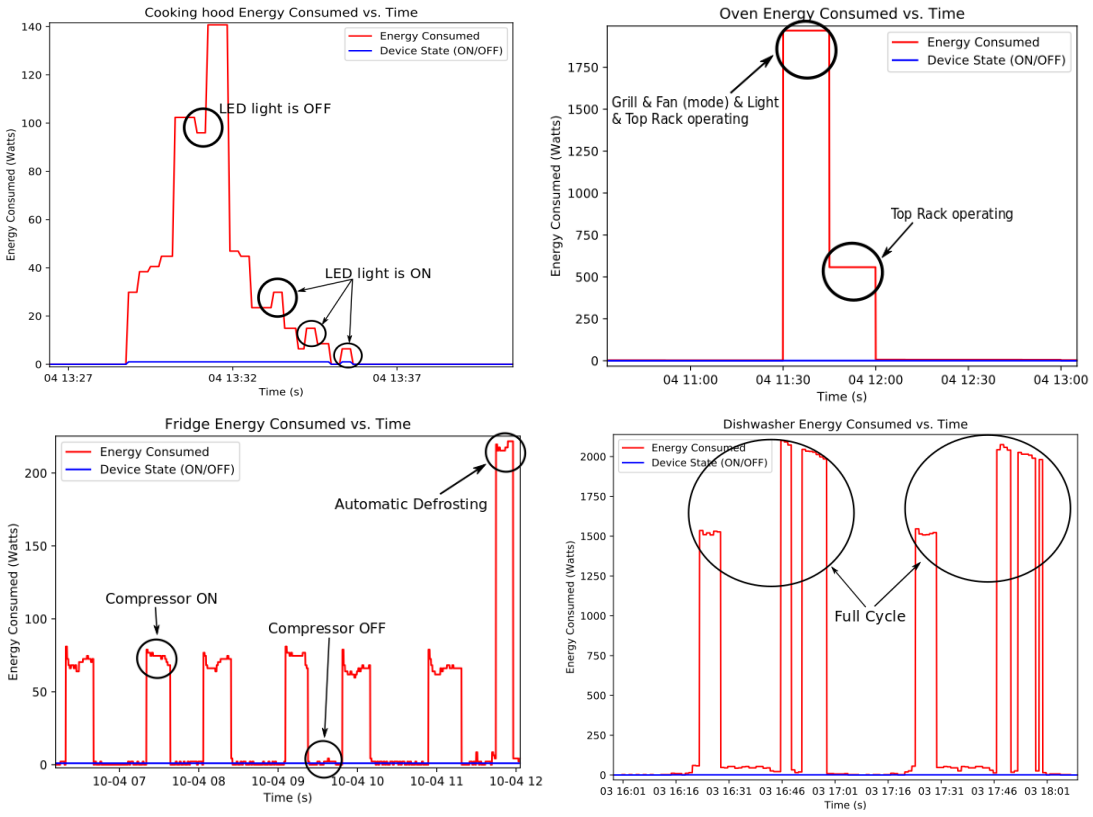


FIGURE 5.3: Power consumption plots of the selected appliances. Top-left is the cooker hood, top-right is the oven, bottom-left is the fridge and bottom-right the dishwasher.

as "OFF" (0) when the reading of the sensor was between 0 and 2.1348 W (a value around 2 W was considered as a 0 for all the appliances from the manufacturer) and "ON" (1) when the reading of the sensor was greater than 3 W. Hence, the dimensions of the overall dataset is 1440(minutes)×4(number of appliances) (for each day, without taking into account the target feature).

Figure 5.3 shows indicative instances of the power consumption for the four appliances from House A. We noticed that we could detect a difference in power consumption regarding the LED state of the cooker hood (measured 4 W). In addition, after measuring the power consumption of the oven (Gavazzi meter sent data every 15 minutes), we could check if there are any "matching" times between the two devices, in order to infer the activity of cooking. Furthermore, the operation of the dishwasher was periodic and therefore quite trivial to infer the activity of washing the dishes. The most challenging appliance was the fridge, since our goal was to detect the appliance usage, in terms of door opening and closing events (based on the fridge light consumption). The fridge located in House A was a state of the art machine, in terms of energy efficiency and consequently it was

not possible to detect when the resident opened and closed the door, even when we increased the data collection time to 20 s.

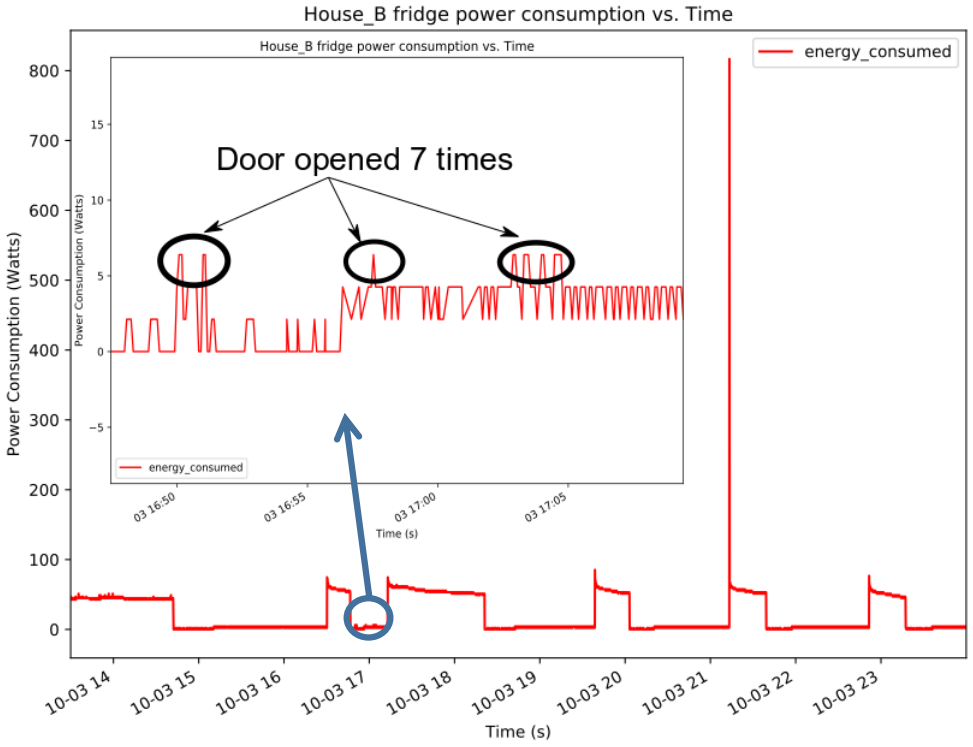


FIGURE 5.4: Fridge power consumption from House B

On the other hand, the fridge that we monitored located in House B was an older model than the one in House A. After sampling at 5 s, we noticed that we could detect when a resident opens and closes the door of the fridge (light turned on) only when the compressor was not operating (Figure 5.4), since the consumption increased from 2 W to 6 W. Otherwise, it was not possible to detect any activity, since there was no difference in the power consumption.

5.2.3 Appliance State Proportion Feature

A key feature to the proposed activity decision engine is the appliance state proportion, which defines the probability of an appliance to be at ON stage on a pre-decided overlapping sliding window [172]. Assume again a set of M activities. In our approach, for each activity $i \in \{1, 2, \ldots M\}$, and each decision time t, a feature $F_t^{(i,j)}$ is calculated for each sensor j, as the proportion of time, or probability, that sensor j is activated at time t, that is:

$$F_t^{(i,j)} = \frac{T_{ON}^{(i,j)}}{T} \tag{5.1}$$

where $0 \leq T_{ON}^{(i,j)} \leq T$ is the total amount of time that sensor j is activated at time t, the latter having a total duration of T. In our work, T was equal to 2 minutes, as it was found to be sufficient for activity detection.

5.3 Decision Engine for Human Activity Recognition

Figure 5.5 depicts our proposed methodology for an energy sensor-based (smart plugs) decision engine for human activity recognition. Assume a set of N sensors (smart plugs) that provide the input data (energy consumption rate of N household appliances) for the N classifiers, where each classifier is dedicated to a specified appliance and $M \geq N$ activities (an activity may relate with more than one appliance). The input data for each classifier is the energy consumption rate measurements of the whole set of appliances while the supervisory signal (used during training) is a vector of the specified appliance states (0 and 1, where 0 denotes the OFF state and 1 denotes the ON state of an appliance). The decision engine calculates the probability of presence of a specific activity using RBS that get as inputs the appliance operating state from the classifier and the appliance state proportion, at the decision time t. An appliance was considered to be active if the state proportion feature was above a threshold of 0.5. In total we collected 21,600 measurements over the 30 days (720 2-minute measurements per day).

We measured the energy consumption of the cooker, cooker hood, oven, washing machine, and dish washer to monitor the three following activities: cooking, washing the dishes, and washing clothes. Therefore for our particular example N was 4 (oven, cooker hood, dishwasher and washing machine) and M was 4 (cooking, washing clothes, washing dishes and doing nothing; when no activity of the aforementioned was performed). Furthermore, we applied a probability boost of 0.3 for the activity of cooking when the cooker hood is ON and a 0.7 for the activity of cooking when the oven is ON, since the state of cooker hood is not related directly with cooking.

A key-feature of the proposed decision engine is the use of an efficient and effective classification technique. To identify a suitable classification technique, we tested

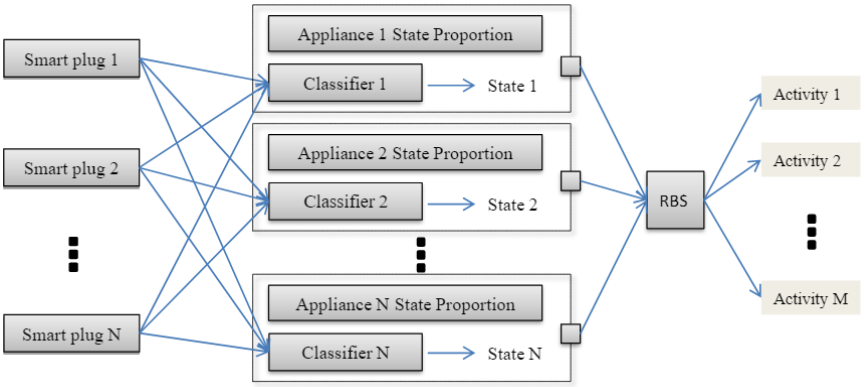


FIGURE 5.5: Overview of energy sensor-based decision engine human activity detection

TABLE 5.1: Accuracy of proposed decision engine

Activities	**Number of correct predictions**	**Ground-truth in 2-minute intervals for 30 days**	**Accuracy (%)**
Cooking	632	685	92.3
Washing Clothes	240	250	96.1
Washing the Dishes	920	965	95.4
Doing Nothing	19,621	19,700	99.6
Average	-	-	**99.1**

the following machine learning methods: SVMs with their basic kernels (Linear, Polynomial and RBF) [173], DT [174], NB [175], LR [176], ANN and specifically a BPN [177]. Along with these standalone machine learning algorithms, the ensemble learning methods of RF [178] and GB [179] were also tested for their predictive performance. We considered the aforementioned classifiers, since a simple thresholding would not be robust against noisy signals and we would lose significant information from the 2 minute-windows.

For the two-class classification scenario of a single household appliance (appliance status OFF/ON), in order to assess our models, the measures of precision, recall, accuracy and MCC were used, which are computed from the contents of the confusion matrix of the classification predictions. Precision is the ratio of predicted true positive cases to the sum of true positives and false positives, recall is the proportion of the true positive cases to the sum of true positives and false negatives and accuracy is the proportion of the total number of predictions that were correct.

Precision or recall alone cannot describe a classifier's efficiency, especially in cases where the labels in training target feature are not balanced. Therefore, MCC is used as balanced evaluation measure and specifically a correlation coefficient among the actual classification and predicted output of the classifier. It returns

a value between -1 and +1, where +1 represents a perfect prediction, 0 random prediction and -1 indicates total disagreement between prediction and observation. MCC is calculated directly from the confusion matrix and is given by the equation:

$$MCC = \frac{TP*TN-FP*FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \tag{5.2}$$

where TP (True Positives), FP (False Positives), TN (True Negatives) and FN (False Negatives).

5.4 Results

We generated 100 Monte Carlo iterations for different parameter scenarios in each classifier to eliminate the bias. For every iteration, we used a random sampling cross-validation where the percentage of samples in the training and the testing datasets was 70% and 30%, respectively. For the SVM with polynomial kernel, the parameter, θ, which is a free parameter taking integer values, is assigned as: θ=(start=30,end=60,step=6) and the polynomial degree takes the values p=(start=2,end=7,step=1). We found that after the 4^{th} degree, we overfitted the dataset. For the SVM with linear kernel, we used the default configuration of scikit-learn [180]. For the SVM with radial basis function kernel, σ varied same as θ and the constant C as C=(start=100,end=1000,step=100). The parameter σ of the RBF kernel handles the non-linear classification. For the DT we used the default optimized version of the classification and regression trees algorithm. For NB we used the Gaussian algorithm for classification and for LR we used the default configuration of scikit-learn. The BPN had a single hidden layer and the number of neurons varies as n=(start=100,end=200,step=20). The RF and GB have an ensemble of *estimators*=(start=20,end=100,step=20) DTs. The combination of all values of parameters and the size of 100 Monte Carlo iterations for each case, results in an overall of 11000 tested cases grid search.

Since more than one classifier achieved the best performance, we selected the SVM with polynomial kernel classifier and performed activity inference for random times. Table 5.1 summarizes the accuracies for the monitored activities (cooking, washing clothes, washing dishes) over 30 days. When an appliance was switched on, our decision engine was not able to instantly detect that it was operating and relate it to an activity. However, after 4 minutes of operation of that particular device it was able to predict the activity related to the operating appliance correctly

and achieved an average accuracy of 99.1%. Furthermore, since the fridge required a high sampling rate in order to determine when the door is open or closed, it is not a device that can be directly related to an activity, such as cooking. However, it can be used as a "supplementary" appliance to increase the confidence of the predicted activity.

5.5 Conclusions

We presented a framework for human activity context inference, based on energy consumption rate from selected appliances. The results are very promising towards unobtrusive activity detection for ambient assisted living. While our experiments were done in a kitchen environment, our approach is flexible enough to be applied to other smart home environments. Since most activities within a house are related with the use of an electrical appliance, this unimodal approach has a significant advantage using inexpensive smart plugs and smart meters for each appliance.

Chapter 6

Investigation of 2D Convolutional Neural Networks

6.1 Introduction

Deep learning techniques such as CNNs have shown good results in activity recognition. One of the advantages of using these methods resides is their ability to generate features automatically. This ability greatly simplifies the task of feature extraction that usually requires domain specific knowledge, especially when using big data where data driven approaches can lead to anti-patterns. Despite the advantage of this approach, very little work has been undertaken on analyzing the quality of extracted features, and more specifically on how model architecture and parameters affect the ability of those features to separate activity classes in the final feature space. The first part of this Chapter focuses on identifying the optimal parameters for recognition of simple activities applying this approach on both signals from inertial and audio sensors.

The second part of this Chapter tackles a NLP problem, using a 2D CNN architecture from the computer vision domain. The main problem in the literature lies in pre-processing of a text (e.g., optical character recognition, word-embedding), in order to convert it to a meaningful input to an RNN or a 1D CNN. We show that the 2D CNNs capture semantically meaningful features from images with text without using optical character recognition techniques and sequential processing pipelines. The 2D CNNs can help develop an end-to-end framework for natural language understanding.

6.2 Comparison of Human Crafted Engineering Features and Learnt Features of a CNN

In this work, the automatic feature extraction of the CNNs and the comparison with the human crafted features are evaluated on large-scale datasets [115, 181].

Regarding the audio modality, the DCASE 2017 development dataset consists of recordings from various acoustic scenes, all having distinct recording locations. For each recording location, 3-5 minute long audio recording was captured. The original recordings were then split into segments with a length of 10 seconds. The total number of recordings were 4680, sampled at 44.1 kHz and were split in four folds (75/25 train/validation split). We compare the recognition accuracy between standard human crafted audio features (MFCCs) and the low-level features that 2D CNNs learn during back-propagation. The MFCCs were selected since they are the most common features used in the fields of speech recognition and environmental sound recognition. Furthermore, since this work used images to train 2D CNNs, it would not be possible to use features such as the zero-crossing rate, spectral centroid, etc.

Regarding the IMU modality, the experiments on the UCI-HAR dataset have been carried out with a group of 30 volunteers within an age bracket of 19-48 years. Each person performed six activities (WALKING, WALKING UPSTAIRS, WALKING DOWNSTAIRS, SITTING, STANDING, LAYING) wearing a smartphone (Samsung Galaxy S II) on the waist. Using its embedded accelerometer and gyroscope, the 3-axial linear acceleration and 3-axial angular velocity at a constant rate of 50Hz were captured.

The contributions focus on a comprehensive analysis on how architecture (number of convolutional layers) and model parameters (size of filters, number of kernels) affect separation of target classes in the feature space. Secondly, in order to verify the selected parameters, a comparison between CNN auto features and gold standard HCF is performed on publicly available datasets. For the inertial sensors, we used the 1D feature vectors from accelerometer and gyroscope. For the audio sensors, we used 2D images, normalized from 0 to 1, of the MFCCs.

The remainder of this Chapter is structured as follows. Section 6.2.1 describes the advantages and disadvantages of automatic feature extraction. Section 6.2.2 describes the approach and the final experiment. The evaluation methodology is

described in Section 6.2.3. Results and discussion are reported in Section 6.2.4 and Section 6.2.5 respectively. Finally, conclusions are drawn in Section 6.2.6.

6.2.1 Automatic Feature Extraction

Among the advantages that DL provides for automatic feature extraction, one of the most relevant is that it does not require domain specific knowledge [105]. In this sense DL provides a standardized way to fulfill the feature extraction step. On the other hand, it introduces some disadvantages. In particular, a training phase is required in order to optimize the weight of the convolutional layers to the characteristic of data from the target domain. This makes DL methods for feature extraction subject to the *cold-start* problem [182], and potentially also to generalization.

The experiment in Section 6.2.3, focuses on feature space rather than on final accuracy. Moreover, the analysis of auto features includes the comparison with gold standard HCF sets.

6.2.2 Approach and Experiment

Typically, CNN architectures include several convolutional layers. In most cases convolutions are followed by a max-pooling operation. The first layer can take directly raw data as input, and the convolutional layers play the role of extracting good features for the final classification. Few cases can be distinguished. For instance, when dealing with inertial sensors (such as accelerometer or gyroscope), the input will be a sequence of samples for each channel. Given an accelerometer signal sampled at 40 Hz, and assuming a window size of 3 s for segmentation, this will produce a 3×120 input in the case all 3 channels (X, Y, Z) are considered separately, or a 1×120 input in the case only the 3D magnitude of the acceleration is taken as input. The input is then processed using temporal convolution (Conv1D) as in [102].

In other cases, 2D convolution is used, as for instance in the case of an image used as input. Audio input typically falls into this category where the 2D magnitude representation of the spectrogram (or any variations of it e.g., mel-spectrogram) or the 2D spectrogram values matrix (e.g., MFCCs) of the audio signal in the time window is used as input. For the case of the MFCC feature extraction, we used

the default sampling rate (44.1 kHz) of the DCASE dataset [181]. The number of MFCCs was 13 (including the $0th$ coefficient), the FFT window size was 2048, with a hop length of 1024 (50% overlap). This resulted in a 13×431 matrix. It has been shown that 2D CNNs outperform 1D CNNs in many audio recognition tasks, since they are able to capture the time-frequency information of the signal [183]. After the sequence of convolutional layers, the output of last convolutional layer is generally flattened into a 1D vector. The output of this step is the automatic feature vector in our experiment.

In the case of a multi-class problem (where classes are mutually exclusive) the output layer is obtained typically using a softmax activation layer in a dense layer. In some cases, a number of dense layers are added in between the flatten and the softmax operation [184], or in case of multi-label classification (where more than one output class can be active at the same time, e.g., "standing" and "walking") other activation functions such as sigmoid can be used [185]. Similarly, the loss function used for training varies commonly from mean squared error in the case of multi-class, or binary cross-entropy in the case of multi-label [185]. The entire process is depicted in Figure 6.1, showing both conventional process with HCF, and automatic features using CNN.

The advantages of the CNN approach are (i) the ability of feeding the model directly with raw data, and (ii) that features are extracted within the series of convolutional operations automatically. On the other hand, as mentioned in Section 6.2.2, the CNN layers will not be able to generate good features, until the model is trained on some known data.

To solve the cold start problem a different dataset can be used for initial training. In our case, one for the IMU feature extractor and the second for the audio signal. Our rationale for the training dataset is to consider a dataset with the following characteristics:

1. the dataset shall not be subject to annotation errors, reducing the risk of overfitting by learning on noisy labels [186]
2. the input data and the target activities must be similar to the target application scenario (so that a feature extractor trained in a similar dataset can be used on the target dataset, in a similar way to a transfer learning approach).

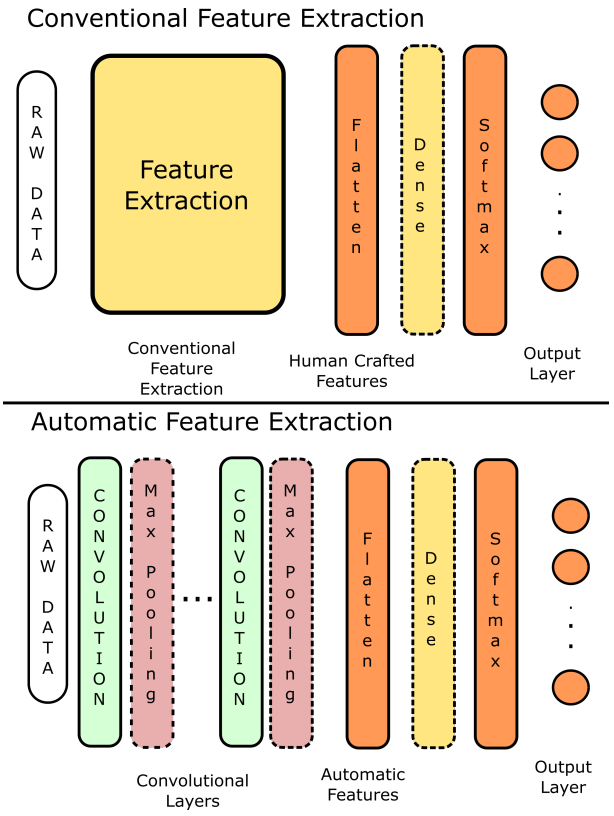


FIGURE 6.1: A typical CNN architecture taking IMU raw data as input will include multiple convolutional layers, with each layer followed by a max-pooling operation. The output of last convolutional layer is then flattened. The vector obtained corresponds to a feature vector automatically extracted. Finally, softmax is generally applied to connect to the output layer in multi-class problems.

In order to fulfill the first requirement, only datasets collected in controlled environment have been considered. Similarly, regarding the second requirement we consider only datasets with the same input data and similar target activity sets.

The initial training phase and the comparison with HCF is performed for the inertial sensor using UCI-HAR dataset [115] and for the audio component the DCASE 2017 development dataset [181]. These two datasets match our aforementioned requirements for training purposes. The main reason for using those datasets is to be able to locate the user.

Figure 6.2 illustrates the training and testing phases of the experiment. UCI-HAR and the DCASE 2017 development dataset are used for training the feature extractor. At this stage the two datasets are used to compare auto features, extracted from the CNNs, with human crafted ones.

Figure 6.2 depicts the proposed process for cross-validation, where the CNN feature extractor is trained on a dataset, then the CNN feature extractor is cross-validated

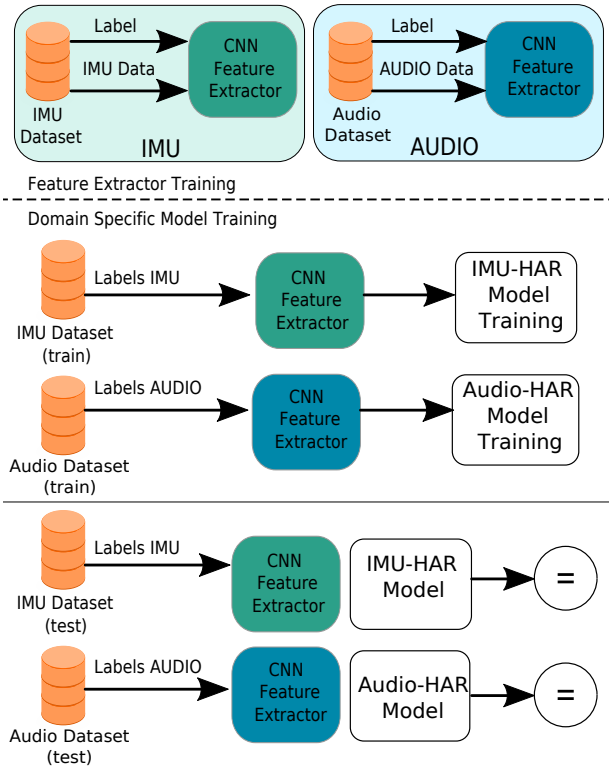


FIGURE 6.2: Cross-validation of a CNN automatic feature extractor: (top) two CNN feature extractors are trained using datasets collected in controlled conditions (UCI-HAR and the DCASE 2017 development datasets), (middle) the CNN model is used as feature extractor, and training data from the final real-world dataset is used to train IMU-HAR and AUDIO-HAR models, (bottom) finally results are evaluated over the final test data.

in conjunction with a classifier model for classification on a different dataset.

6.2.3 Evaluation

In the experimental work two datasets have been used, UCI-HAR dataset [115] for inertial sensors, and the DCASE 2017 development dataset [181] for the audio case. For the audio case, we simplified the 15 classes to 3, based on the location (indoor, outdoor and vehicle). Simplifying the classes helps in a scenario where inertial and audio sensors would be used to approximate the user's location, without a GPS sensor. UCI-HAR dataset fulfills our requirements being a dataset collected under controlled conditions. IMU data have been collected using a Samsung Galaxy S II smartphone, placed on the waist. The dataset includes data from 30 participants and provides benchmark training (70% of participants) and test dataset (30% participants). This type of evaluation enables accuracy performance to be tested on users that have not been part of the training. The dataset provides

a benchmark set of 348 features, extracted from time and frequency domains of the accelerometer signal. This benchmark set has been used as the set of HCF in the comparison.

6.2.3.1 Extracted Features

The aforementioned datasets were used to train the feature extractor, and the features obtained were compared to human crafted ones. The comparison aims also at verifying how parameters affect the quality of automatically generated features. The analysis investigated the following parameters:

1. number of convolutional layers
2. kernel size used for convolution

The final accuracy of the model was complemented with plots, visualizing how activities were separated in the feature space, both in the case of HCF and automatic features. The visual comparison assists to analyze and interpret results obtained in terms of accuracy and provides a more complete evaluation of CNN features.

6.2.3.2 Experimental Environment

The experimental framework was implemented using Python. A Keras [140] implementation of CNN was used, with TensorFlow [165] as the backend.

6.2.4 Results

This Section describes the results that were obtained regarding the IMU modality on the UCI-HAR dataset and the acoustic modality on the DCASE 2017 Task 1 development dataset.

6.2.4.1 IMU CNN Features

Regarding automatic generation of CNN features for the IMU, the experiment focused on the set of target activities defined in the UCI-HAR [115] dataset (i.e.,

'Laying', 'Sitting', 'Standing', 'Walking', 'Walking Upstairs' and 'Walking Downstairs'). Feature quality has been measured on accuracy performances of CNN models using different layers of convolution (n-CNN where n is the number of layers) and different values of kernel size k. Models were trained for 150 epochs with a batch size of 512 samples. The benchmark train and test sets from UCI-HAR have been used. The training set has been further split using 90% for training and 10% as validation for early stop criterion to avoid overfitting. The UCI-HAR dataset provides IMU raw data for the accelerometer and gyroscope signal. Data are split in segments with a window size of 128 samples and 50% overlap, obtained from IMU signals sampled at 50 Hz. The accelerometer signal is separated into gravity and body components, separated using a Butterworth low-pass filter (cut-off 0.3 Hz). The separation makes a total of 9 channels, 6 for the accelerometer (X,Y,Z for both gravity and body components) and 3 for the gyroscope (X,Y,Z). Consequently, an input of 128×9 or 128×6 was obtained by taking accelerometer and gyroscope signals, or accelerometer only. For multi layer CNN models, the number of filters used were 12, 24, 48 and 96 respectively, each followed by a max-pooling operation. For IMU 1D convolution was used. A flatten layer was used after the last convolutional, followed by a dense layer (64 nodes and relu activation function), and the final output with the 6 classes using a softmax and adam optimizer with learning rate lr=0.001. The comparison with manually engineered features was performed using the same architecture as the CNN case after the flatten layer. The input layer would take HCF (a 348-dimensional features vector using accelerometer only, and 561 with both accelerometer and gyroscope) with a dense layer of 64 nodes, and the final output layer. Comparison between HCF and CNN automatic features was performed on the same architecture as in Figure 6.3.

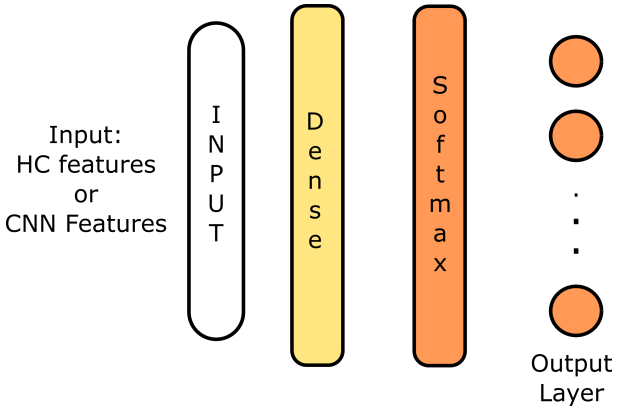


FIGURE 6.3: Architecture used to compare HCF and CNN features: taking features in input, one dense layer (64 nodes) and 6 classes (for the inertial sensors) and 3 classes (for the audio sensors) on the output layer.

Figure 6.4 (left column) depicts a visualization of the feature space obtained performing PCA, reducing to three dimensions for visualization purposes. To facilitate visual inspection of data points separation in the feature space, in Figure 6.4-6.5 the plot has been obtained excluding data points labeled as 'Laying', which were far away in the feature space from all other activities; including them would affect interpretation of visualized data. Training of models using different kernel sizes has been performed. Figure 6.4 (right column) depicts data points in the feature space for activities using varying kernel sizes. Figure 6.5 depicts visual

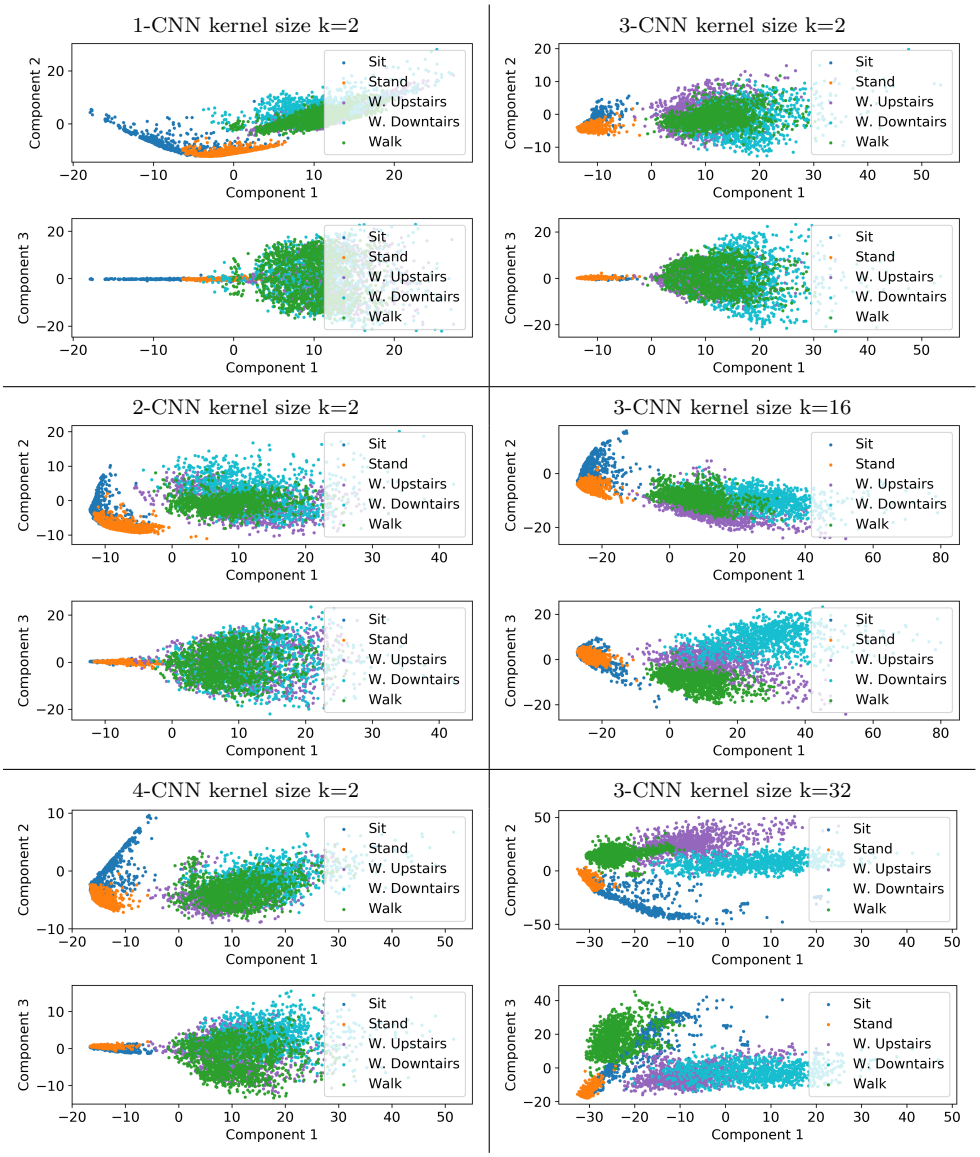
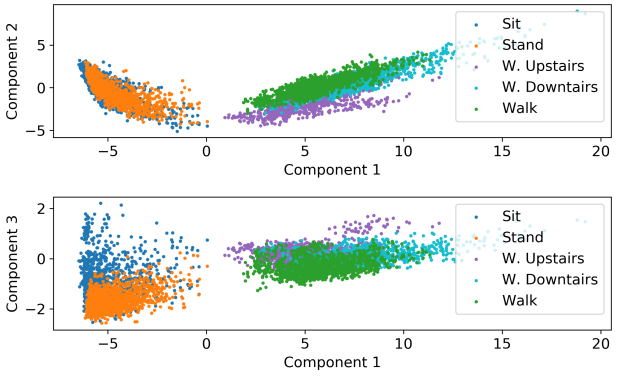


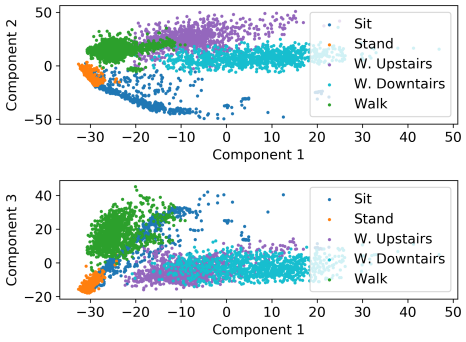
FIGURE 6.4: Visual comparison of 1D CNN architectures using $n = 1, 2$ and 4 layers with a kernel $k = 2$ (on the left), and kernel size $k = 2, 16, 32$ using $n = 3$ layers (on the right).

comparison of target activities separation in the feature space using (top) HCF,

and (bottom) automatic features obtained with 3-CNN model and using kernel size $k = 32$.



(a)



(b)

FIGURE 6.5: Visual comparison of (a) HCF, and (b) CNN using 3 layers.

Table 6.1 and 6.2 summarizes average precision, recall and F-score with the tested models, using accelerometer only, and both accelerometer and gyroscope signals.

Figure 6.6 provides further insight on how error rates were distributed between classes, comparing HCF with a CNN model (3 layers kernel size $k = 64$). The confusion between sitting and standing can be explained by the fact that the human torso is oriented the same way during both activities.

6.2.4.2 Audio CNN Features

Regarding automatic generation of CNN for the DCASE 2017 development dataset, the experiment focused on grouping the 15 given classes (beach, bus, cafe/restaurant, car, city center, forest path, grocery store, home, library, metro station, office, park, residential area, train and tram) to three classes, namely outdoor, indoor and vehicle. The raw spectrogram images were used as the input to the CNN. In order to extract the spectrogram of the signal, an FFT size of 512 with a hop

TABLE 6.1: Precision, Recall and F-Score obtained on UCI-HAR Dataset using HCF and CNN features obtained with different parameters using accelerometer only.

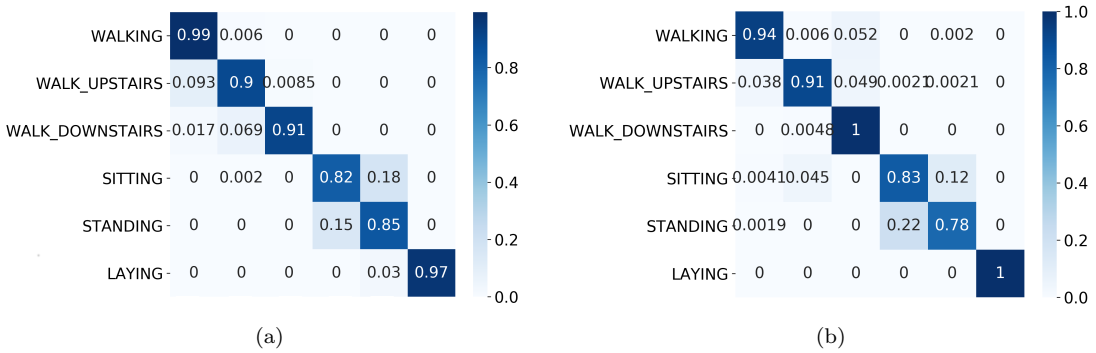
Parameters	Precision	Recall	F-Score
HCF (acc only)[a]	**89.95%**	**89.38%**	**89.58%**
1-CNN K=2	85.31%	84.63%	84.41%
2-CNN K=2	88.26%	87.95%	87.96%
3-CNN K=2	90.73%	90.57%	90.55%
4-CNN K=2	89.62%	89.21%	89.19%
3-CNN K=8	**90.55%**	**90.26%**	**90.20%**
3-CNN K=16	90.71%	90.09%	90.16%
3-CNN K=32	88.24%	87.89%	87.87%
3-CNN K=64	88.17%	87.95%	87.97%

[a]set of 348 accelerometer only.

TABLE 6.2: Precision, Recall and F-Score obtained on UCI-HAR Dataset using HCF and CNN features obtained with different parameters using accelerometer and gyroscope.

Parameters	Precision	Recall	F-Score
HCF (acc & gyro)[a]	**95.80%**	**95.39%**	**95.50%**
1-CNN K=2	88.84%	89.01%	88.87%
2-CNN K=2	89.54%	89.70%	89.59%
3-CNN K=2	90.51%	90.63%	90.55%
4-CNN K=2	91.84%	91.97%	91.89%
3-CNN K=8	91.44%	91.63%	91.51%
3-CNN K=16	92.96%	93.08%	92.98%
3-CNN K=32	**93.31%**	**93.52%**	**93.38%**
3-CNN K=64	92.03%	92.20%	92.04%

[a]561 features: accelerometer and gyroscope.



(a) (b)

FIGURE 6.6: Normalized confusion matrices obtained using (a) HCF and (b) using CNN.

length of 512 was used. Furthermore, the original recording was down-sampled to 16 kHz. The reason for down-sampling and using a non-overlapping window for FFT was due to the length of the recording (10 s), which would produce an image size that could not be processed. Therefore, the resulting image after the pre-processing step was 257×313 pixels. Feature quality has been measured on accuracy performances of CNN models using different layers of convolution (n-CNN where n is the number of layers) and different values of kernel size k. The filters used for each CNN layer were 32, 48, 120 and 120 respectively followed by a 2×2 max-pooling layer. The CNNs were trained between 20-30 epochs (for different folds and different network sizes) and the selected batch size was 32. The number of epochs was selected based on the early stopping criterion, in order to avoid overfitting. The ReLU [167] activation function was used for each convolutional and max-pooling layer and the Adam [136] optimizer was used to train the networks with an initial learning rate $lr = 0.001$.

For our experiments we used the default 4-fold cross validation that is provided in [181]. However, we show the PCA analysis for the second fold, since it was the most challenging one. The other folds follow a similar trend. Table 6.3 summarizes average precision, recall and F-score with the tested models. We notice that the best model consisted of 2 convolutional layers with a kernel size of 2. Figure 6.7 (left column) depicts a visualization of the feature space obtained performing PCA, reducing to three dimensions for visualization purposes. Figure 6.7 (right column) depicts data points in the feature space for activities using varying kernel sizes. The variance of the PCA components increases as the number of kernels increases. The first and third principal components show that for the case of two convolutional layers (kernel of size 2), two classes can be distinguished in the feature space.

6.2.5 Discussion

6.2.5.1 IMU CNN Features

Results obtained using different number of layers of convolution highlights how a model with 3-4 layers outperforms models with 1 or 2 layers in F-score. At the same time, increasing the number of convolutional layers does not improve accuracy, but only increases the complexity of the model. The results are confirmed with the visualization of data points in Figure 6.4, showing how a 4 layer model better separates activities in the feature space. Smaller values of kernel size correspond

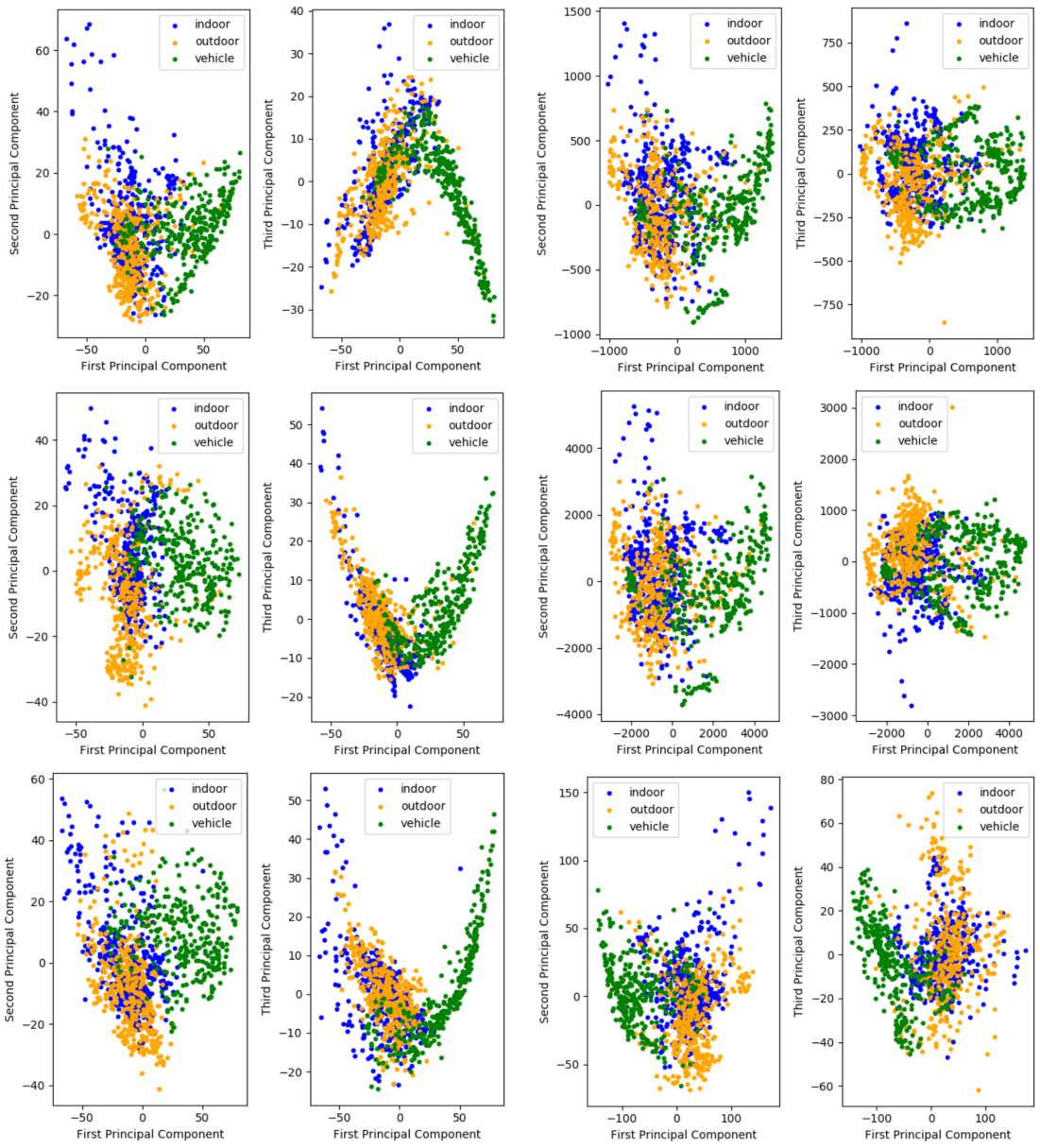


FIGURE 6.7: Visual comparison (first with second PCA and first with third PCA) of 2D CNN architectures using $n = 1$ (top left), $n = 2$ (mid left) and $n = 4$ (bottom left) layers with a kernel $k = 2$, and kernel size $k = 8$ (top right), $k = 16$ (mid right), $k = 32$ (bottom right) with $n = 2$ layers for the audio dataset.

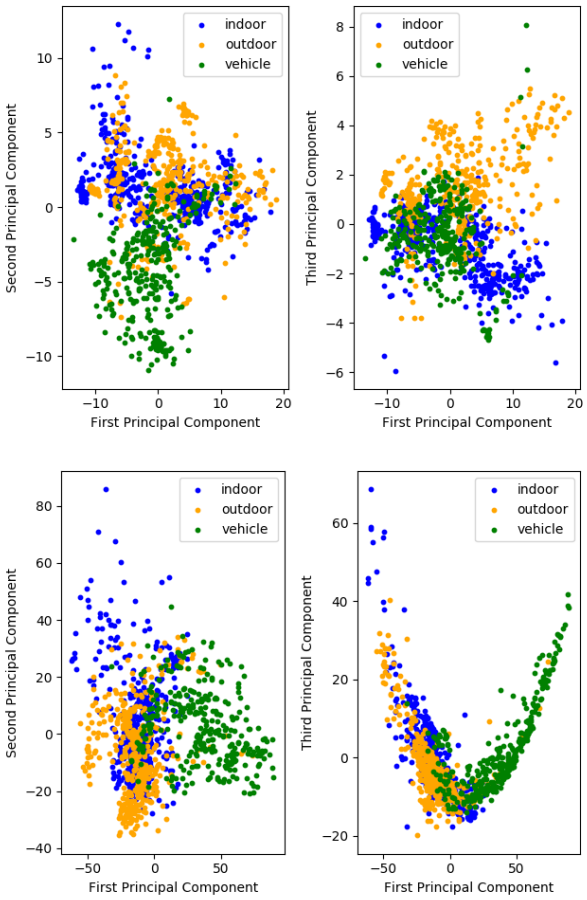
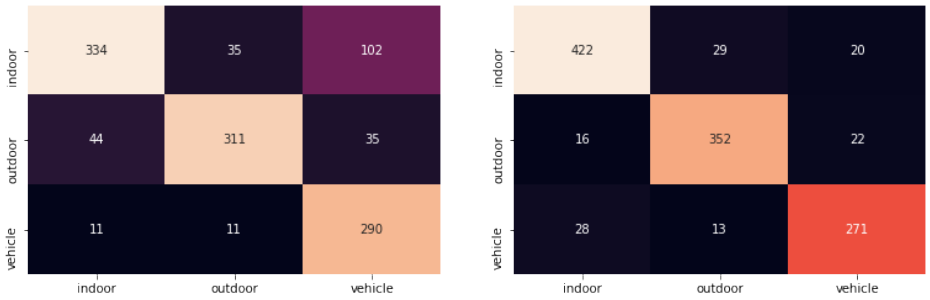


FIGURE 6.8: Visual comparison of HCF (top) and CNN using 2 layers (bottom) for the audio sensor.

TABLE 6.3: Precision, Recall and F-Score (averaged over 4-folds) obtained on the DCASE 2017 development using different parameters.

Parameters	**Precision**	**Recall**	**F-Score**
HCF (MFCCs)	85.37%	85.22%	84.75%
1-CNN K=2	51.63%	60.47%	53.72%
2-CNN K=2	**91.02%**	**90.2%**	**90.5%**
3-CNN K=2	90.9%	90.17%	90.45%
4-CNN K=2	90.14%	89.56%	89.74%
2-CNN K=8	89.1%	88.62%	88.78%
2-CNN K=16	49.58%	52.9%	52.12%
2-CNN K=32	11.09%	33.33%	16.59%
2-CNN K=64	12.23%	33.33%	17.86%



(a) (b)

FIGURE 6.9: Un-normalized confusion matrices obtained using (a) HCF and (b) using CNN for the audio dataset.

to lower accuracy values; conversely increasing the kernel size over 32, decreased accuracy. When using larger kernel sizes, 3 and 4 layers provide similar results, thus a 3 layers approach is preferable since it reduces model's complexity. It should be noted that data segmentation in this dataset has been performed with window size of approximately 2.5 seconds. With a sampling rate of 50 Hz the best performing kernel sizes (8 and 16) corresponds to 0.3 and 0.6 seconds approximately. Summarizing, increasing the number of layers helps to better separate inter-group variability between static (sitting, standing) and active labels (walking, walking upstairs and walking downstairs). On the other hand, increasing the kernel size helps to better separate data points intra-group for both active and static labels. The insight provided with visualization is confirmed by recognition performance of models measured using precision, recall and F-score. Results confirm that CNN automatic features are able to provide accuracy performances comparable with best known set of HCF, and are in line with performances measured in [102]. In this work, classification using only the accelerometer has also been evaluated. In this case CNN features provided higher precision and recall compared to HCF. When considering both accelerometer and gyroscope, HCF provide about 1-2% higher F-score, although that is including frequency domain features.

6.2.5.2 Audio CNN Features

Good recognition accuracy can be obtained using only two convolutional layers, followed by max-pooling. For the 2D CNN architectures, increasing the kernel size, while keeping a relatively shallow network (two layers), decrease the recognition accuracy performance. The network performance would increase by stacking more convolutional layers, thus increasing the complexity of the model. Furthermore, experiments show that the kernel size of the 2D CNN should be small, in order

to capture all the details in the time and frequency domain. Figure 6.8 shows that a 2-layer 2D CNN can distinguish the three target classes after being trained for 22 epochs from raw spectrogram images. The top part of the figure depicts the human crafted MFCC features. When visualizing the first and third principal components we notice that there is not a clear distinction between the classes. Confusion matrices in Figure 6.9 show that the CNN can outperform HCF for indoor and outdoor settings. However, for the selected dataset, HCF achieved better classification accuracy in the vehicle environment. This probably occurred since the MFCCs are not robust to noisy environments. The results were promising, especially for training deep networks on device. To ensure privacy of sensitive audio data, networks should be light-weight and able to capture data and adapt (re-train) on the embedded system, ensuring no information is stored on the cloud.

6.2.6 Conclusions

In this work, an analysis of performance of CNN extracted features has been presented. The experiment focused on comparison of automatically extracted and HCF for activity recognition. In particular, the audio signal, accelerometer and gyroscope data have been investigated. Moreover, the effect of important parameters has been evaluated, namely number of convolutional layers, and kernel size used for the convolution. Automatically extracted features achieved comparable results with the HCF on the UCI-HAR dataset. Furthermore, the automatically extracted features of the 2D CNN from the raw-spectrogram outperformed the HCF on the DCASE 2017 development dataset. On the one hand, it must be considered that using CNN features provides a standard way for feature extraction, simplifying the process compared to the human crafted case. On the other hand, a CNN used as feature extractor requires an initial training phase in order to generate good features (cold-start problem). The experiments provide insight on CNN feature performances; however, further work should evaluate performance of CNN, on large real-world datasets. Next steps will include experiments cross-validating a pre-trained CNN feature extractor on different datasets, with different sets of target activities.

6.3 2D Convolutional Neural Networks for dialogue modelling

An important part of the ACROSSING project that sponsored this PhD, was the collaboration between the Early Stage Researchers. An end goal of the ACROSSING project was the creation of a virtual assistant that would be able to interact with the user regarding providing recipes or checking any abnormalities (e.g., calling a relative of the user when no activity is detected for one day). To achieve this goal, the 2D CNNs used for the audio-based event detection were chosen to solve the NLP task of dialogue modeling. Although one would argue that the results of the audio-based event detection could be used as a text file for further processing, the purpose of this research was to examine if the same approach of using a two-dimensional image of a text (including timestamps with activities in a tabular format), as in the case of the spectrograms, could provide comparable classification results with state-of-art 1D neural networks. The two-dimensional CNNs would be able to remove the need for OCR, leading towards an end-to-end system.

Recent advances in NLP make heavy use of neural network models. Solutions for tasks such as semantic tagging [187], text classification [188] and sentiment analysis [189] rely on either RNN or CNN variants. In the latter case, the vast majority of the proposed models are based on character-level CNNs applied on one-hot vectors of text or 1D CNNs [190]. Although the results are promising, having either surpassed or equaled the previous state of the art, there are a few issues regarding the proposed models, which are all related to the fundamental inductive bias underlying these models' architectural design. Whether working at the word- or character-level, language processing with most neural network models almost always translates to sequential processing of a string of abstract discrete symbols.

CNNs based on 1D or character convolutions constitute the vast majority of CNN models used in language processing. These networks are fast if the dictionary size is small. However, for some languages, the one-hot encoding vector dimension for input sequences can be very large (e.g., over 3,000 for Chinese characters). Furthermore, and specifically for RNN variants, training for long input sequences is difficult due to the well-known problem of vanishing gradients. While architectures like LSTM [191] and GRU [192] were specifically designed to tackle this problem,

stable training on long sequences remains an elusive goal, with recent works devising yet more ways to improve performance in recurrent models [193, 194, 195]. Moreover, many state of the art recurrent models rely on the attention mechanism to improve performance [196, 197, 198], which places an additional computational burden on the overall method.

To tackle the above problems, we use CNNs to process the entire text at once as an image. In other words, we convert our textual datasets into images of the relevant documents and apply our model on raw pixel values. This allows us to sensibly apply 2D convolutional layers on text, taking advantage of advances in neural network models designed for and targeting computer vision problems. Doing so, allows us to bypass the issues stated earlier relating to the use of 1D character-level CNNs and RNNs, since now the processing of documents relies on parallel extraction of visual features of many lines (depending on filter size) of text. Regarding the vanishing gradient problem, we can take advantage of recent CNN architectural advances [199, 200, 201], which specifically aim to improve its effects. In terms of linguistics, our approach is based on the distributional hypothesis [202], where our model produces compositional hierarchies of document semantics by way of its hierarchical architecture. Beyond providing an alternative computational method to deal with the problems described above, our approach is also motivated by findings in neuroscience, cognitive science and the medical sciences where the link between visual perception and recognition of words and semantic processing of language has long been established [203, 204]. Our approach is robust to textual anomalies, such as spelling mistakes, unconventional use of punctuation (e.g., multiple exclamation marks), etc. which factors in during feature extraction. As a result, not only is the need of laborious text preprocessing removed, but the derived models are able to capture the semantic significance of the occurrence of such phenomena (e.g., multiple exclamation marks to denote emphasis), which proves to be especially helpful in tasks such as text classification and/or sentiment analysis. Moreover, our approach can work with any text (latin and non-latin), text font, misspellings and punctuation. Furthermore, it can be extended to handwriting, background colors and table formatted text naturally. It also removes the need of pre-processing real-world documents (and thus the need for optical character recognition, spell check, stemming, and character encoding correction).

The proposed approach is based on the hypothesis that more semantic information can be extracted from features derived from the visual processing of text than by processing strings of abstract discrete symbols. We test this hypothesis on

NLP tasks and show that a solid capture of text semantics leads to better model performance. Our contributions are summarized as follows:

- a proof of concept that text classification can be achieved over an image of the text;
- a proof of concept that basic dialogue modeling (restaurant booking), in an information retrieval setting, can be completed using only image processing methods;

The remainder of the Chapter is organized as follows: Section 6.3.1 positions our approach compared to related work, Section 6.3.2 introduces the proposed method, Section 6.3.3 presents the experimental results and Section 6.3.4 draws the conclusions.

6.3.1 Related Work

The use of convolutional neural networks for natural language processing has attracted increasing attention in recent years. For sentence classification, Kim [205] used a simple CNN architecture consisting of one convolutional layer with multiple filters of different sizes, followed by max-pooling. The feature maps produced are then fed to a softmax layer for classification. Despite its simplicity, this architecture exhibited good performance. Sentence modeling was further explored by Blunsom et al. [206] who used an extended application, which they call Dynamic CNN to deal with various input lengths and short- and long-term linguistic dependencies. Wang et al. [207] perform clustering in an embedding space to derive semantic features which they then feed to a CNN with a convolutional layer, followed by k-max pooling and a softmax layer for classification.

Character-level (as opposed to word- or sentence-level) feature extraction was investigated by Zhang et al. [208] who used a standard deep convolutional architecture for text classification. Conneau et al. [209] showed that using very deep convolutional architecture improves results over standard deep convolutional networks on text classification tasks. Dos Santos and Gatti [210] carried out sentiment analysis on sentences taken from text corpora, using a CNN architecture which derives input representations that are hierarchically built from the character to the sentence level. Johnson and Zhang [211] used a CNN for text categorization. Their

method does not rely on pre-trained word embeddings, but rather computes convolutions directly on high-dimensional text data represented by one-hot vectors. An architectural variation was also proposed for adapting a bag-of-words model in the convolutional layers. Johnson and Zhang [212] used CNNs for sentiment and topic classification in a semi-supervised framework, where they retained the representations derived by a CNN over text regions, and which they then integrated into the supervised CNN classifier. Ruder et al. [213] employed a novel architecture combining character- and word-level channels to determine an unseen text's author among a large number of potential candidates, a task they called large-scale authorship attribution. Bjerva et al. [214] introduced a semantic tagging method, which combines stacked neural network models and a residual bypass function. The stacked neural networks consist of a vanilla CNN or a ResNet [200] in the lower level for character-/word-level feature extraction and a bidirectional GRU in the higher level. The residual bypass function preserves the saliency of lower-level features that could be potentially lost in the processing chain of intermediate layers.

Dialogue managers can be trained either as generative models or as discriminative models to differentiate good replies in NUC [215]. In generative models [216, 217, 218], dialogue managers are trained to produce replies for a given dialogue history. In NUC setting, a dialogue manager needs to choose the correct response from a set of candidate replies as Memory Networks (MemNets) [219, 220] in Facebook bAbI dataset [221].

While all the aforementioned works exploited CNNs for NLP tasks, they all used text data as input, either pre-trained word embeddings or simply one-hot vector representations.

6.3.2 Proposed Method

In our approach, we treat text classification as a problem which concerns the learning of context-dependent semantic conventions of language use in a given corpus of text. We treat this complex problem as an image processing problem, where the model processes an image with the text body (Figure 6.10), learning both the local (word- and sentence-level) and the global semantics of the corpus. In this way, the domain or context dependent meaning of sentences is implicitly contained in the variations of the visual patterns given by the distribution of words in sentences. As such, the problem is that the model needs to observe as

many variations of in-domain text as possible to be able to generalize adequately. This process is similar to the analytical method of learning to read [222], where the global meaning of a body of text is acquired first and learning of the text's meaning moves to hierarchically lower linguistic units. In our case, this translates to capturing the structure and context of the whole corpus first, then the sentences, and finally the words that constitute these sentences.

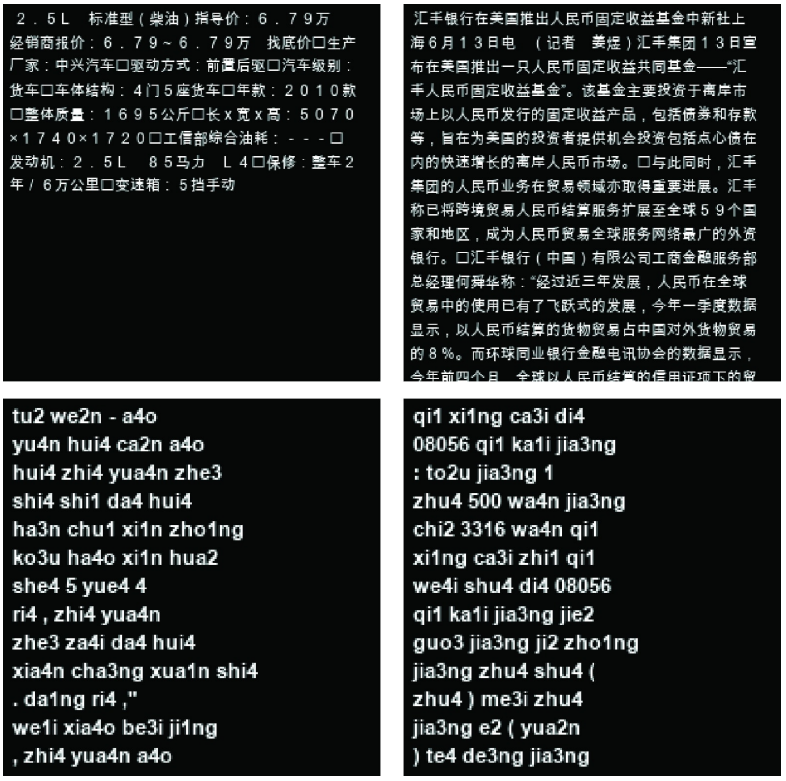


FIGURE 6.10: Top: Sogou News dataset with Chinese characters. Bottom: Sogou News dataset with pinyin (romanization of the Chinese characters based on their pronunciation)

6.3.2.1 Models

For the tasks of (English and Chinese) text classification we used a vanilla CNN and also the Xception architecture [201] to check whether better vision deep networks can increase performance.

The vanilla CNN consists of seven convolutional layers, a fully connected layer and an output layer containing as many units as classes (e.g., for a classification problem with four classes, the output layer would contain four units). All filters in the convolutional layers are 5×5 with stride 2. The first three layers use 32 filters, while the rest use 64 filters. The fully connected layer consists of 128 units.

All units in all layers use the rectifier function, apart from the output layer, which uses a softmax output. Figure 6.11 shows the architecture of the model.

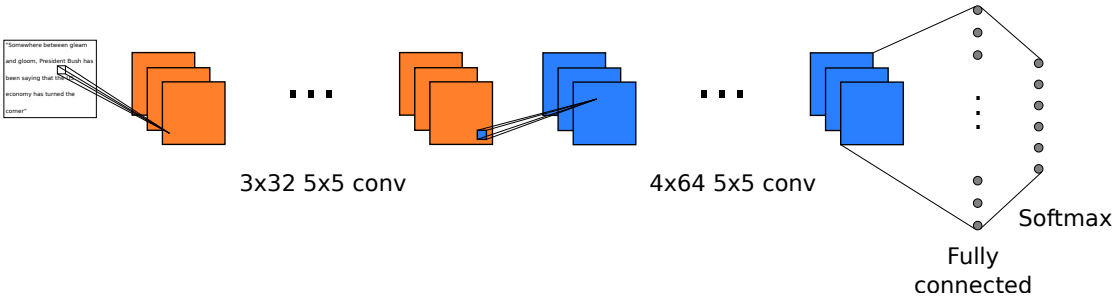


FIGURE 6.11: Proposed model: 3 convolutional layers consisting of 32 filters with a kernel of size 5×5 each, are followed by 4 convolutional layers consisting of 64 filters with a kernel of size 5×5 each. A linear fully connected layer and a classification output layer complete the model.

For the task of dialogue modeling we used version 4 of the recently proposed deep Inception network (Inception-V4) [223]. Our choice was motivated by the fact that the vanilla CNN model was too simple to effectively model the dialogue structure, as well as its pragmatics (i.e., the use of language in discourse within the context of a given domain), a problem which Inception-V4 seems to have tackled, at least to a certain extent. We selected the Inception-V4 against the Xception because it is a lighter network in terms of training times and provides robust recognition accuracy results in many tasks.

6.3.2.2 Data Augmentation

Data augmentation has been shown to be essential for training accurate models [209, 224]. For image recognition, augmentation is applied using simple transformations such as shifting the width and the height of images by a certain percentage, scaling, or randomly extracting sub-windows from a sample of images [225].

For the task of English and Chinese text classification, we used the *ImageDataGenerator* function provided by Keras [140]. The input image was shifted in width and height by 20%, rotated by 15 degrees and flipped horizontally, using a batch size of 50. For the task of dialogue modeling, we applied the same augmentation techniques and random character flipping. Character flip and in particular changing the rating of a restaurant improved the per-response and per-dialogue accuracy, especially for difficult sentences, such as booking a 4 star restaurant.

6.3.3 Results

To validate our approach, we ran experiments for two separate tasks: text classification and dialogue modeling, using a single NVIDIA GTX 1080 Ti GPU.

6.3.3.1 Text classification

In this task we trained our model on an array of datasets which contained text related to news (AG's News and Sogou's News), structured ontologies on Wikipedia (DBPedia), reviews (Yelp and Amazon) and question answering (Yahoo! answers). Details about the datasets can be found in [208]. For this task, Zhang et al. [208] tested CNNs that use 1D convolutions in the task of text classification, which may more broadly include natural language processing, as well as sentiment analysis. While the model in [208] uses text as input vectors, our proposed method uses image data of text. In other words, whereas Zhang et al. [208] use one-hot vector representations of words or word embeddings, we use binarized pixel values of grayscale images of text corpora.

TABLE 6.4: Results of Latin and Chinese text classification in terms of held-out accuracy. Worst-Best Performance reports the results of the worst and best performing baselines from Table 4 of Zhang et al. [208] and Conneau et al. [209]. Results reported for *TI-CNN* were obtained in 10 epochs

Dataset	**Worst-best Performance (%)**	**TI-CNN (%)**	**Xception (%)**	**Number of Classes**
AG's News	83.1-92.3	80.0	91.8	4
Sogou News (Pinyin)	89.2-97.2	90.2	94.6	5
Sogou News (Chinese)	93.1-94.5	-	**98.0**	5
DBPedia	91.4-98.7	91.7	94.5	14
Yelp Review Polarity	87.3-95.7	90.3	92.8	2
Yelp Review Full	52.6-64.8	55.1	55.7	5
Yahoo! Answers	61.6-73.4	57.6	73.0	10
Amazon Review Full	44.1-63	50.2	57.9	5
Amazon Review Polarity	81.6-95.7	88.6	94.0	2

Table 6.4 shows our method's held-out accuracy in the task of Latin and Sogou News in Chinese text classification for each of the datasets. All baselines are derived from Table 4 of Zhang et al. [208] and Conneau et al. [209]. We denote the vanilla CNN by *TI-CNN* (Text-to-Image Convolutional Neural Networks). The column *Worst-Best Performance* shows the worst and best held-out accuracy achieved by the baseline models. Our approach achieved comparable results to most of the best performing baselines. The Amazon datasets were large and we

TABLE 6.5: Generated text for testing. The following text samples were not seen during the training

Sample No	Text Sample	Positivity Score	Sample No	Text Sample	Positivity Score
1	this product is mediocre	0.60	5	I love this product it is great	0.99
2	this product is excelent	0.91	6	I like this product it is ok	0.78
3	this product is excellent	0.96	7	I don't know	0.56
4	this product is excellent!!!	0.98	8	as;kdna;sdn nokorgmnsd kasdn;laknsdnaf	0.51

did not have enough computational resources to achieve comparable results to the state-of-art with Xception.

Table 6.5 shows human generated text (not included in the training set) used for testing. For these examples, the table shows predictions after the model was trained on the *Amazon Review Polarity* dataset [226], which contains reviews of products in various product categories. The dataset is used for binary (positive/negative) sentiment classification of text and the metric *(positivity score)* is the probability of the positive class. The model was able to discriminate between words expressing different degrees of the same sentiment (e.g., samples 1,6 compared to samples 2-5). Sample 2 (compared to samples 3-4) illustrates our method's robustness to anomalies like spelling mistakes. In a traditional NLP setting the misspelled word would have a different representation from the respective correctly-spelled word. Unless the model was trained on data that contained many of these anomalies, or engineered by a human, it would not necessarily correlate the misspelled word with the sentiment it expressed. In our model the misspellings are handled naturally. We note that while this can be alleviated by preprocessing procedures or character-level models, these require more pre-processing or human intervention than our method.

As discussed before, the model builds these visual representations in a bottom-up fashion, creating a semantic hierarchy which is derived from language use within the context of the corpus domain. Sample 4 shows another interesting characteristic of our model which is capturing the effect of punctuation (exclamation marks) even if used informally. The exclamation marks used in sample 4 generated the highest prediction score for positive sentiment among all variations of the same phrase (samples 2-4). Samples 5 and 6 have a similar structure but the different choice of words to describe positive sentiment affects the prediction score. This also exhibits the model's capacity to build meaningful hierarchical representations, as it has learned to discriminate between the small nuances (e.g., choice of words)

encountered in (visually and semantically) similar textual structures (sentences). Interestingly, an input which expresses a "neutral" sentiment, such as sample 7, has an analogous prediction score (0.56) that is closer to random guessing in a model that was trained in binary sentiment prediction, which is reasonable behavior. The model is also robust to nonsensical text such as sample 8.

Finally, we applied the Xception architecture to the Sogou News dataset, using the original Chinese characters (Figure 6.10). Huang and Wang [190] used 1D CNNs for text classification with Chinese characters and showed that the accuracy recognition was higher than the traditional conversion to the pinyin romanization system. We extended this work by using the Xception architecture in the 2D image to achieve almost the same result (Table 6.4). This proves that regardless of how many Chinese words we fit in a 300×300 or a 200×200 image, our approach outperformed the NLP sequential CNNs. Furthermore, the performance improved when using the Chinese characters instead of the pinyin.

6.3.3.2 Dialogue modeling

For the dialogue modeling task, we tested our Inception-V4-based agent in task 4 of the bAbI dialogue dataset [221], since it requires knowledge base information when choosing the replies to the user (e.g., address, phone number). The bAbI dialogue dataset consists of 1000 training, 1,000 validation and 1,000 test dialogues in the restaurant booking domain. Each dialogue is divided in four different tasks. Here we focus on task 4, where the dialogue agent should be able to read entries about restaurants from a relevant knowledge base and provide the user the requested information, such as restaurant address or phone number. We note that restaurant phone numbers and addresses have been delexicalized and replaced by tokens representing this information. We chose to focus on this task to demonstrate the increased effectiveness of visual processing of dialogue as opposed to purely linguistic processing, due to the high number of different lexical tokens. In our approach the agent needs to correlate the visual pattern of a knowledge base entry to the relevant request. While in principle this should be easy to achieve using artificial delexicalized tokens, as in this benchmark task, it would be far more difficult to do so in the real world, with non-standard sequences of words (such as restaurant names, addresses etc). However, given the results of the text classification tasks, we hypothesize that given enough data, our visual approach can create semantic models that encapsulate such correlations.

As in text classification, we trained the model with images of dialog text taken from the bAbI corpus. So the agent learns the expected user utterances and their corresponding responses on the system side by processing images of in-domain dialogue text. The agent learned visual representations of text meaning and structure both at word-level (implicitly, through the optimization process) and utterance-level (explicitly, through labeling of correct and incorrect responses given a user utterance).

TABLE 6.6: Facebook bAbI Dialogue Task 4

Metrics	**Inception-V4 (%)**	**Memory Networks w/o Match Type (%)**
Per-response Accuracy	63.3	59.5
Per-dialogue Accuracy	11.4	3.0

Table 6.6 shows the Inception-V4 performance against the MemNets used in [221]. The table shows that, our approach is competitive with MemNets when the latter does not use match types. Bordes et al. [221] introduced match types to make their model rely on type information, rather than exact match of word embeddings corresponding to words that frequently appear as containing OOV entities (restaurant name, phone number). This is because it is hard to distinguish between similar embeddings in a low-dimensional embedding space (e.g., phone numbers) as they lead to full scores in all metrics. In real life, match types would require a lexical database to identify every word type which is not realistic.

6.3.4 Conclusions

We presented a proof of concept that natural language processing can be based on visual features of text. For non-dialogue text, images of text as input to CNN models can build hierarchical semantic representations which let them detect various subtleties in language use, irrespective of the language of the input data. For dialogue text, we showed that CNN models learn both the structure of discourse and the implied dialogue pragmatics implicitly contained within the training data. Although our model is trained in an NUC setting, it could be expanded as a generative model by using an image-based encoder for dialogue history and a language-based model for decoding. Crucially, unlike traditional NLP applications, our approach does not require any preprocessing of natural language data, such as tokenization, optical character recognition, stemming, or spell checking. Our method can work using different computer fonts, background colors and can

be expanded to human handwriting. It can perform NLP tasks on real-word documents that include tables, bold, underlined and colored text, where traditional NLP methods, as well as language agnostic models (1D CNN) fail.

Our work is a first step towards expanding the methods for natural language processing, exploiting recent advances in image recognition and computer vision. Initial results of this approach are promising for a wide range of NLP tasks, such as text classification, sentiment analysis, dialogue modeling and natural language processing.

Chapter 7

Conclusions

7.1 Summary of Contributions

This thesis has studied the task of using machine learning algorithms for human activity detection in domestic environments, with a particular focus on acoustic and energy consumption sensors. The human activities vary for each user and their home environment and we have advanced the research in the cases of examining different environments using the aforementioned modalities. We have also investigated an end-to-end system for speech activity detection, as human speech plays an important role in activity recognition and needs to be separated from the other environmental sounds. Below is a summary of the contributions made by this thesis:

- Developed two frameworks for audio-based event detection and examined the statistical significance of between traditional classifiers and a CNN architectures (Section 3.3.1).
- Identified the statistical significance of well-known audio features for the problem of audio-based event detection in a kitchen environment, in the presence of background noise (Section 3.3.2).
- Demonstrated the effect of the duration of the signal segment used for classification on recognition accuracy. Decreasing the segment duration decreases the response time of the system but may harm its recognition accuracy. At the same time, increased duration can lead to increased co-occurrence of multiple events within the same sound segment (Section 3.3.3).

- Investigated the SNR and the distance between the microphone and how does the audio source affect the recognition accuracy in a new environment (i.e., one which was not used to train the classifier) (Sections 3.3.4 and 3.3.5).

- Proposed an end-to-end 1D CRNN and a 2D CRNN, using spectrogram images as input, for speech/non-speech activity detection. In both systems, we calculated convolutions of 10 ms windows (80 samples) average to correct speech predictions ranging from 0.01 to 0.5 s (Chapter 4)

- Presented a framework for unobtrusive human activity context inference, based on energy consumption rate from selected appliances and using a decision engine based on operation of the appliances (Chapter 5).

- Investigated the performance of CNN extracted features. The experiments focused on comparison of automatically extracted and HCF for activity recognition. In particular, the audio signal, accelerometer and gyroscope data have been investigated. Moreover, the effect of important parameters has been evaluated, namely the number of convolutional layers, and the kernel size used for the convolution (Section 6.2) .

- Presented a proof of concept that natural language processing can be based on visual features of text. For non-dialog text, images of text as input to CNN models can build hierarchical semantic representations which let them detect various subtleties in language use, irrespective of the language of the input data. For dialogue text, we showed that CNN models learn both the structure of discourse and the implied dialogue pragmatics implicitly contained within the training data. This work was inspired by building an end-to-end dialogue assistant in a domestic environment. The audio was transformed to STFT spectrogram images and the recognized event from the audio-based framework was transformed from a text to an image. The two images were trained using the same network and this lead to the reduction of number of trainable parameters as well as training time per epoch (Section 6.3).

7.2 Limitations of this Work

Despite obtaining promising results for most of the problems studied in this thesis there are still some limitations of this work. These include:

- The inability of audio-based event detection systems to distinguish between overlapping events. Since only one acoustic sensor was used, only the loudest event was identified. For instance, when the microphone was placed 6 m away from the mixer and at the same time 3 m away from the kitchen faucet, the systems were able to correctly classify only the activity of the mixer, since the sound of the mixer masked entirely the sound of the running tap water. The same limitation applies to multiple people doing multiple activities (multi-label scenario).

- Training the proposed audio networks requires a powerful desktop PC. When deploying the system on a single board computer, one has to stream data to the desktop server in order to initiate the training. Therefore, for the final solution the network has to be optimized so the training can run on device (e.g., single board computer) and avoid the client/server application.

- Activities that are not related to electrical appliances, such as sleeping and taking a shower. A single modality for human activity recognition based on the active power consumption is not sufficient for those cases. Therefore, a combination of other modalities (e.g., audio) is needed.

- Deployment infrastructure of the energy-based activity inference method. When analyzing many appliances in a home environment, one would require to setup a smart meter on every appliance and an MQTT broker for communication. This could cause data communication problems between the broker and the database, especially when capturing data every minute, resulting in a sparsity matrix from the missing data

- Answering the question whether the SAD system could generalize by separating speech coming from the speakers of the television and live speech in an indoor environment. The proposed framework has been trained with background noise coming from specific audio sources and would be prone to any similar sounds from other audio sources (e.g., sound of cicadas can be confused with the running water from a faucet).

- The ability of the 2D CNNs on natural language understanding when varying the size of the input image. Currently the system expects a specific image size in pixels and only certain words can fit in it. Furthermore, it can only work with computer fonts and not human alphabet ones.

- Testing the human activity detection framework, using the audio modality, on a new real-world dataset that has not been included in the training.

7.3 Open Issues and Future Work

Although the problem of single audio-based event detection was researched, robust polyphonic event detection in different environments remains an open problem. Various room effects (e.g., reverberation and echo) can affect the recognition accuracy of a developed system. Although many solutions have been proposed, future work could focus on reinforcement learning techniques that can penalize the false positives and false negatives when detecting the starting time of an audio event.

Additionally, public audio datasets are increasing in number and in size (big data). Manually labeling them, in order to obtain the ground-truth, can be a tedious task, since the labels are prone to human error but also expensive when professionals are called to label the sounds. Specifically in the case of environmental sounds where there are many similarities in the case of a water boiling sound and the sound of a coffee pot that can be related to cooking and preparing coffee, respectively. Regarding this problem, unsupervised algorithms (e.g., autoencoders) have been proposed to use the latent space that is "learnt" from the network for clustering the audio event. A possible extension to these unsupervised methods could be a deep hierarchical variational autoencoder, which uses a mixture of discrete and continuous distributions to learn to effectively separate the different data manifolds and would be trainable end-to-end. This method could work for real-world datasets with significant class imbalance, where we the discriminative capabilities of a purely unsupervised framework.

Regarding an energy-based event detection system, one of the important challenges is to test the system in a new "untrained" home environment, where appliances such as the fridge, dishwasher and washing machine will have a different profile of power consumption. Additionally, our work assumed that the dataset was seamlessly disaggregated. This is not realistic for most domestic environments, meaning that one cannot disaggregate from the main power consumption appliances such as a laptop or a desk lamp, but rather focus on appliances that consume more power (e.g., oven and microwave). Therefore, there is a strong need of accurate disaggregation that can work robustly in different domestic environments. A possible solution to improve the performance of disaggregation algorithms is by using spectrogram images, of the active power consumption, as input to 2D CNN denoising autoencoders, since the 2D CNNs can identify strong patterns in the time and frequency domains.

For the problem of SAD, developing algorithms that are robust in recognition accuracy in both quiet and noisy environments and are able to generalize well under unseen environments remains an unsolved problem. In a multi-condition environment, such as the Apollo-11 dataset, single channels have different SNRs. Therefore, in this supervised problem, a possible solution to improve the proposed algorithm is to apply the 2D CRNN to each channel and average the scores. Additionally, pseudo-labeling, test time shift augmentation techniques and varying the signal length can significantly affect the performance of the SAD system.

Generalization of a developed neural network architecture is a broad open problem for all the modalities that were researched in this thesis. Particularly, the image-based text classification using 2D CNNs can work well with computer fonts but has not been tested with human fonts. Similarly in the case of using IMU and audio sensors for human activity inference, a particular network architecture can work well for one or two datasets. The problem of generalization could be solve by using advanced deep neural networks, however, this would make the system impossible to be deployed in a mobile device. Another solution could be to focus on pre-trained models that could be used simply to initialize the weights, perform ensemble of different architectures either by majority voting or fuse the layers earlier before the fully connected layer.

7.4 Concluding Remarks

This thesis presented the use of machine and deep learning algorithms for human activity recognition in indoor environments. We specifically focused on audio sensors and the single event detection in an indoor environment. The study of the effect of audio features and classifiers can be used by researchers and practitioners in their task of selecting the most suitable features for their classifier.

Speech is a very common part of the every day life. Separating the speech sound signals from the other sound sources is important, especially for home assistant devices (e.g., Amazon Alexa, Google Home), since it can contain sensitive information for each user. We believe that we advanced the research in the field of speech activity detection by focusing on the post-processing of an audio signal and not the neural network architecture itself. Even though there is a great interest in complex deep neural networks that can achieve the highest possible accuracy, with very simple post-processing steps (e.g., convolutions with ones to act as a high-pass filter), we can still increase the recognition accuracy.

Using smart power consumption meters, the activity of a user in a home environment can be inferred using traditional machine learning classifiers with one type of feature vector (active power consumption) and showed that we should not always focus on deep neural network architectures.

Since most of the CNN architectures are considered as "black-boxes", simple CNN architectures were investigated in order to identify a criterion of selection of number of kernels and filter sizes for particular datasets.

Finally, this research approached two NLP tasks from a different perspective than what is common in the literature (e.g., 1D CNNs, word-embedding with attention RNNs) and showed that we can outperform state-of-the-art architectures in Latin and non-Latin alphabets.

Appendix A

Ethics Approval AKTIOS

ELDERLY CARE UNITS

Aktios & Aktios Alzheimer Unit
17 Ipirou st., Agia Paraskevi, GR 15341
Tel.: +30 210 6515268 • info@aktios.gr

Aktios Odigos
157 Varis Ave., Vari, GR 16672
Tel.: +30 210 8971413 • odigos@aktios.gr

Aktios Art Palace
55 Pentelis st., Maroussi, GR 15126
Tel.: +30 210 8022912 • artpalace@aktios.gr

www.aktios.gr

Athens, 30/6/2017

Decision of the Ethics Committee

Today, June 30 2017, the Ethics Committee of AKTIOS SA held a session in Athens, Greece. The meeting was about the research with the title "Automatic detection of daily activities using acoustic data" and its methodology.

The document for approval was the following:

1. Research methodology : "Automatic detection of daily activities using acoustic data"

The Committee after a thorough review approved the document.

The Ethics Committee of AKTIOS was formed in 2017 and is an advisory board that is designated by the board of directors. It consists of healthcare professionals that are working at AKTIOS Elderly Care Units and its role is to check that all the research projects, in which AKTIOS SA participates, meet the requirements of good research practice in terms of ethics and the code of conduct.

The Committee consists of five (5) people:

President:

Dr. Costis Prouskas

Members:

Dr. Kimon Volikas

Dr. Vasilios Liras

Mr. Efstathios Trifonopoulos

Mr. Anestis Karipidis

30/06/2017

The President of the Committee

Costis Prouskas, Ph.D.

Psychologist – Gerontologist

Bibliography

[1] J. Nehmer, M. Becker, A. Karshmer, and R. Lamm, "Living assistance systems: An ambient intelligence approach," in *Proceedings of the 28th International Conference on Software Engineering*, ser. ICSE '06. New York, NY, USA: ACM, 2006, pp. 43–50.

[2] N.-C. Chi and G. Demiris, "A systematic review of telehealth tools and interventions to support family caregivers," *Journal of Telemedicine and Telecare*, vol. 21, no. 1, pp. 37–44, 2015.

[3] L. Pérez-Lombard, J. Ortiz, and C. Pout, "A review on buildings energy consumption information," *Energy and buildings*, vol. 40, no. 3, pp. 394–398, 2008.

[4] P. Rashidi and A. Mihailidis, "A survey on ambient-assisted living tools for older adults," *IEEE journal of biomedical and health informatics*, vol. 17, no. 3, pp. 579–590, 2013.

[5] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, "Activity recognition using cell phone accelerometers," *ACM SigKDD Explorations Newsletter*, vol. 12, no. 2, pp. 74–82, 2011.

[6] C. Duarte, K. Van Den Wymelenberg, and C. Rieger, "Revealing occupancy patterns in an office building through the use of occupancy sensor data," *Energy and Buildings*, vol. 67, pp. 587–595, 2013.

[7] A. Jalal, Y.-H. Kim, Y.-J. Kim, S. Kamal, and D. Kim, "Robust human activity recognition from depth video using spatiotemporal multi-fused features," *Pattern recognition*, vol. 61, pp. 295–308, 2017.

[8] A. Vafeiadis, K. Votis, D. Giakoumis, D. Tzovaras, L. Chen, and R. Hamzaoui, "Audio-based event recognition system for smart homes," in *Ubiquitous*

Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld), 2017 Intl IEEE Conferences. IEEE, 2017.

[9] G. Bauer, K. Stockinger, and P. Lukowicz, "Recognizing the use-mode of kitchen appliances from their current consumption." *EuroSSC*, vol. 9, pp. 163–176, 2009.

[10] V. Peltonen, J. Tuomi, A. Klapuri, J. Huopaniemi, and T. Sorsa, "Computational auditory scene recognition," in *Acoustics, speech, and signal processing (icassp), 2002 IEEE international conference on*, vol. 2. IEEE, 2002, pp. II–1941.

[11] K. Yatani and K. N. Truong, "Bodyscope: a wearable acoustic sensor for activity recognition," in *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*. ACM, 2012, pp. 341–350.

[12] D. Yu and L. Deng, *Automatic Speech Recognition*. Springer, 2016.

[13] S. Gatehouse, G. Naylor, and C. Elberling, "Benefits from hearing aids in relation to the interaction between the user and the environment," *International Journal of Audiology*, vol. 42, no. sup1, pp. 77–85, 2003.

[14] A. Mesaros, T. Heittola, A. Eronen, and T. Virtanen, "Acoustic event detection in real life recordings," in *Signal Processing Conference, 2010 18th European*. IEEE, 2010, pp. 1267–1271.

[15] T. Heittola, A. Mesaros, T. Virtanen, and A. Eronen, "Sound event detection in multisource environments using source separation," in *Machine Listening in Multisource Environments*, 2011.

[16] F. Vesperini, L. Gabrielli, E. Principi, and S. Squartini, "Polyphonic sound event detection by using capsule neural networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 2, pp. 310–322, 2019.

[17] J. Z. Kolter and M. J. Johnson, "Redd: A public data set for energy disaggregation research," in *Workshop on data mining applications in sustainability (SIGKDD), San Diego, CA*, vol. 25, no. Citeseer, 2011, pp. 59–62.

[18] M. Maasoumy, B. Sanandaji, K. Poolla, and A. S. Vincentelli, "Berdsberkeley energy disaggregation data set," in *Proceedings of the Workshop*

on Big Learning at the Conference on Neural Information Processing Systems (NIPS), 2013, pp. 1–6.

[19] S. Gupta, M. S. Reynolds, and S. N. Patel, "Electrisense: single-point sensing using emi for electrical event detection and classification in the home," in *Proceedings of the 12th ACM international conference on Ubiquitous computing*, 2010, pp. 139–148.

[20] H. Mshali, T. Lemlouma, and D. Magoni, "Adaptive monitoring system for e-health smart homes," *Pervasive and Mobile Computing*, vol. 43, pp. 1–19, 2018.

[21] A. D. Wood, J. A. Stankovic, G. Virone, L. Selavo, Z. He, Q. Cao, T. Doan, Y. Wu, L. Fang, and R. Stoleru, "Context-aware wireless sensor networks for assisted living and residential monitoring," *IEEE network*, vol. 22, no. 4, 2008.

[22] M. Gietzelt, K. Wolf, M. Kohlmann, M. Marschollek, R. Haux *et al.*, "Measurement of accelerometry-based gait parameters in people with and without dementia in the field," *Methods Inf Med*, vol. 52, no. 4, pp. 319–325, 2013.

[23] M. Marschollek, A. Rehwald, K. Wolf, M. Gietzelt, G. Nemitz, H. M. Zu Schwabedissen, and R. Haux, "Sensor-based fall risk assessment–an expert 'to go'," *Methods of information in medicine*, vol. 50, no. 05, pp. 420–426, 2011.

[24] L. Palmerini, S. Mellone, G. Avanzolini, F. Valzania, and L. Chiari, "Quantification of motor impairment in parkinson's disease using an instrumented timed up and go test," *IEEE transactions on neural systems and rehabilitation engineering*, vol. 21, no. 4, pp. 664–673, 2013.

[25] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors." *IEEE Communications Surveys and Tutorials*, vol. 15, no. 3, pp. 1192–1209, 2013.

[26] L. Chen, J. Hoey, C. D. Nugent, D. J. Cook, and Z. Yu, "Sensor-based activity recognition," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 6, pp. 790–808, 2012.

[27] L. Chen, C. Nugent, and G. Okeyo, "An ontology-based hybrid approach to activity modeling for smart homes," *IEEE Transactions on human-machine systems*, vol. 44, no. 1, pp. 92–105, 2014.

[28] Z.-Y. He and L.-W. Jin, "Activity recognition from acceleration data using ar model representation and svm," in *Machine Learning and Cybernetics, 2008 International Conference on*, vol. 4. IEEE, 2008, pp. 2245–2250.

[29] A. M. Khan, Y.-K. Lee, S. Lee, and T.-S. Kim, "Accelerometer's position independent physical activity recognition system for long-term activity monitoring in the elderly," *Medical & biological engineering & computing*, vol. 48, no. 12, pp. 1271–1279, 2010.

[30] A. M. Khan, Y.-K. Lee, S. Y. Lee, and T.-S. Kim, "A triaxial accelerometer-based physical-activity recognition via augmented-signal features and a hierarchical recognizer," *IEEE transactions on information technology in biomedicine*, vol. 14, no. 5, pp. 1166–1172, 2010.

[31] T. Plötz, N. Y. Hammerla, and P. Olivier, "Feature learning for activity recognition in ubiquitous computing," in *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, vol. 22, no. 1, 2011, pp. 1729–1734.

[32] J. Yin, Q. Yang, and J. J. Pan, "Sensor-based abnormal human-activity detection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 8, pp. 1082–1090, 2008.

[33] M. Buettner, R. Prasad, M. Philipose, and D. Wetherall, "Recognizing daily activities with rfid-based sensors," in *Proceedings of the 11th international conference on Ubiquitous computing*. ACM, 2009, pp. 51–60.

[34] J. Wu, A. Osuntogun, T. Choudhury, M. Philipose, and J. M. Rehg, "A scalable approach to activity recognition based on object use," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. IEEE, 2007, pp. 1–8.

[35] S. T. M. Bourobou and Y. Yoo, "User activity recognition in smart homes using pattern clustering applied to temporal ann algorithm," *Sensors*, vol. 15, no. 5, pp. 11 953–11 971, 2015.

[36] N.-C. Chi and G. Demiris, "A systematic review of telehealth tools and interventions to support family caregivers," *Journal of Telemedicine and Telecare*, vol. 21, no. 1, pp. 37–44, 2015.

[37] Y. Lu, Y. Wei, L. Liu, J. Zhong, L. Sun, and Y. Liu, "Towards unsupervised physical activity recognition using smartphone accelerometers," *Multimedia Tools and Applications*, vol. 76, no. 8, pp. 10 701–10 719, 2017.

[38] L. Meng, C. Miao, and C. Leung, "Towards online and personalized daily activity recognition, habit modeling, and anomaly detection for the solitary elderly through unobtrusive sensing," *Multimedia Tools and Applications*, vol. 76, no. 8, pp. 10 779–10 799, 2017.

[39] A. Bulling, U. Blanke, and B. Schiele, "A tutorial on human activity recognition using body-worn inertial sensors," *ACM Computing Surveys (CSUR)*, vol. 46, no. 3, p. 33, 2014.

[40] Y. Chen and C. Shen, "Performance analysis of smartphone-sensor behavior for human activity recognition," *IEEE Access*, vol. 5, pp. 3095–3110, 2017.

[41] A. J. Perez and S. Zeadally, "Privacy issues and solutions for consumer wearables," *IT Professional*, vol. 20, no. 4, pp. 46–56, 2018.

[42] P. Kumari, L. Mathew, and P. Syal, "Increasing trend of wearables and multimodal interface for human activity monitoring: A review," *Biosensors and Bioelectronics*, vol. 90, pp. 298–307, 2017.

[43] D. Giakoumis, G. Stavropoulos, D. Kikidis, M. Vasileiadis, K. Votis, and D. Tzovaras, "Recognizing daily activities in realistic environments through depth-based user tracking and hidden conditional random fields for mci/ad support," in *European Conference on Computer Vision*. Springer, 2014, pp. 822–838.

[44] I. Kostavelis, D. Giakoumis, S. Malassiotis, and D. Tzovaras, "Human aware robot navigation in semantically annotated domestic environments," in *International Conference on Universal Access in Human-Computer Interaction*. Springer, 2016, pp. 414–423.

[45] J. Chen, A. H. Kam, J. Zhang, N. Liu, and L. Shue, "Bathroom activity monitoring based on sound," in *International Conference on Pervasive Computing*. Springer, 2005, pp. 47–61.

[46] M. Vacher, D. Istrate, F. Portet, T. Joubert, T. Chevalier, S. Smidtas, B. Meillon, B. Lecouteux, M. Sehili, P. Chahuara *et al.*, "The sweet-home project: Audio technology in smart homes to improve well-being and reliance," in *Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE*. IEEE, 2011, pp. 5291–5294.

[47] I. Bisio, A. Delfino, F. Lavagetto, M. Marchese, and A. Sciarrone, "Gender-driven emotion recognition through speech signals for ambient intelligence

applications," *IEEE Transactions on Emerging Topics in Computing*, vol. 1, no. 2, pp. 244–257, 2013.

[48] K.-Y. Huang, C.-C. Hsia, M.-s. Tsai, Y.-H. Chiu, and G.-L. Yan, "Activity recognition by detecting acoustic events for eldercare," in *6th World Congress of Biomechanics (WCB 2010). August 1-6, 2010 Singapore*. Springer, 2010, pp. 1522–1525.

[49] M. Sehili, B. Lecouteux, M. Vacher, F. Portet, D. Istrate, B. Dorizzi, and J. Boudy, "Sound environment analysis in smart home," *Ambient Intelligence*, pp. 208–223, 2012.

[50] A. Temko, C. Nadeu, D. Macho, R. Malkin, C. Zieger, and M. Omologo, "Acoustic event detection and classification," in *Computers in the human interaction loop*. Springer, 2009, pp. 61–73.

[51] H. Lozano, I. Hernáez, A. Picón, J. Camarena, and E. Navas, "Audio classification techniques in home environments for elderly/dependant people," in *International Conference on Computers for Handicapped Persons*. Springer, 2010, pp. 320–323.

[52] J. Chen, A. H. Kam, J. Zhang, N. Liu, and L. Shue, "Bathroom activity monitoring based on sound," in *Proceedings of the Third International Conference on Pervasive Computing*, ser. PERVASIVE'05. Berlin, Heidelberg: Springer-Verlag, 2005, pp. 47–61.

[53] F. Kraft, R. Malkin, T. Schaaf, and A. Waibel, "Temporal ICA for classification of acoustic events in a kitchen environment," in *INTERSPEECH, Lisbon, Portugal*, 2005.

[54] R. M. Alsina-Pagès, J. Navarro, F. Alías, and M. Hervás, "homesound: Real-time audio event detection based on high performance computing for behaviour and surveillance remote monitoring," *Sensors*, vol. 17, no. 4, p. 854, 2017.

[55] M. Vacher, B. Lecouteux, P. Chahuara, F. Portet, B. Meillon, and N. Bonnefond, "The sweet-home speech and multimodal corpus for home automation interaction," in *The 9th edition of the Language Resources and Evaluation Conference (LREC)*, 2014, pp. 4499–4506.

[56] S. Chu, S. Narayanan, and C.-C. J. Kuo, "Environmental sound recognition with time–frequency audio features," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 6, pp. 1142–1158, 2009.

[57] D. Barchiesi, D. Giannoulis, D. Stowell, and M. D. Plumbley, "Acoustic scene classification: Classifying environments from the sounds they produce," *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 16–34, 2015.

[58] G. Parascandolo, T. Heittola, H. Huttunen, T. Virtanen *et al.*, "Convolutional recurrent neural networks for polyphonic sound event detection," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 6, pp. 1291–1303, 2017.

[59] H. Eghbal-Zadeh, B. Lehner, M. Dorfer, and G. Widmer, "CP-JKU submissions for DCASE-2016: A hybrid approach using binaural i-vectors and deep convolutional neural networks," *IEEE AASP Challenge on Detection and Classification of Acoustic Scenes and Events (DCASE)*, 2016.

[60] J. Liu, X. Yu, W. Wan, and C. Li, "Multi-classification of audio signal based on modified svm," in *IET International Communication Conference on Wireless Mobile and Computing (CCWMC 2009)*, Dec. 2009, pp. 331–334.

[61] Y. Xu, Q. Huang, W. Wang, P. Foster, S. Sigtia, P. J. B. Jackson, and M. D. Plumbley, "Unsupervised feature learning based on deep models for environmental audio tagging," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 6, pp. 1230–1241, Jun. 2017.

[62] J. Li, W. Dai, F. Metze, S. Qu, and S. Das, "A comparison of deep learning methods for environmental sound detection," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Mar. 2017, pp. 126–130.

[63] J. Portelo, M. Bugalho, I. Trancoso, J. Neto, A. Abad, and A. Serralheiro, "Non-speech audio event detection," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on.* IEEE, 2009, pp. 1973–1976.

[64] A. J. Eronen, V. T. Peltonen, J. T. Tuomi, A. P. Klapuri, S. Fagerlund, T. Sorsa, G. Lorho, and J. Huopaniemi, "Audio-based context recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 1, pp. 321–329, 2006.

[65] J. T. Geiger, B. Schuller, and G. Rigoll, "Large-scale audio feature extraction and svm for acoustic scene classification," in *Applications of Signal Processing to Audio and Acoustics (WASPAA), 2013 IEEE Workshop on*. IEEE, 2013, pp. 1–4.

[66] F. Fuhrmann, A. Maly, C. Leitner, and F. Graf, "Three experiments on the application of automatic speech recognition in industrial environments," in *International Conference on Statistical Language and Speech Processing*. Springer, 2017, pp. 109–118.

[67] H. Erdogan, J. R. Hershey, S. Watanabe, and J. Le Roux, "Deep recurrent networks for separation and recognition of single-channel speech in non-stationary background audio," in *New Era for Robust Speech Recognition*. Springer, 2017, pp. 165–186.

[68] I. McLoughlin, H. Zhang, Z. Xie, Y. Song, W. Xiao, and H. Phan, "Continuous robust sound event classification using time-frequency features and deep learning," *PloS one*, vol. 12, no. 9, 2017.

[69] C.-Y. Wang, J.-C. Wang, A. Santoso, C.-C. Chiang, and C.-H. Wu, "Sound event recognition using auditory-receptive-field binary pattern and hierarchical-diving deep belief network," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 8, pp. 1336–1351, 2018.

[70] A. S. Bregman, *Auditory scene analysis: The perceptual organization of sound*. MIT press, 1994.

[71] D. P. W. Ellis, "Prediction-driven computational auditory scene analysis," Ph.D. dissertation, Massachusetts Institute of Technology, 1996.

[72] Y. Zhang and W. H. Abdulla, "Gammatone auditory filterbank and independent component analysis for speaker identification," in *Ninth International Conference on Spoken Language Processing*, 2006.

[73] D. L. Wang and G. J. Brown, "Separation of speech from interfering sounds based on oscillatory correlation," *IEEE transactions on neural networks*, vol. 10, no. 3, pp. 684–697, 1999.

[74] S. N. Patel, T. Robertson, J. A. Kientz, M. S. Reynolds, and G. D. Abowd, "At the flick of a switch: Detecting and classifying unique electrical events on the residential power line (nominated for the best paper award)," in *International Conference on Ubiquitous Computing*. Springer, 2007, pp. 271–288.

[75] F. Rosenblatt, "Recent work on theoretical models of biological memory," *JT Tou (Ed.)*, 1967.

[76] H. Lee, P. Pham, Y. Largman, and A. Y. Ng, "Unsupervised feature learning for audio classification using convolutional deep belief networks," in *Advances in neural information processing systems*, 2009, pp. 1096–1104.

[77] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[78] Y. Kim, H. Lee, and E. M. Provost, "Deep learning for robust feature generation in audiovisual emotion recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 3687–3691.

[79] L. Deng, J. Li, J.-T. Huang, K. Yao, D. Yu, F. Seide, M. Seltzer, G. Zweig, X. He, J. Williams *et al.*, "Recent advances in deep learning for speech research at microsoft," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 8604–8608.

[80] Y.-H. Tu, J. Du, Q. Wang, X. Bao, L.-R. Dai, and C.-H. Lee, "An information fusion framework with multi-channel feature concatenation and multi-perspective system combination for the deep-learning-based robust recognition of microphone array speech," *Computer Speech & Language*, vol. 46, pp. 517–534, 2017.

[81] K. J. Piczak, "Environmental sound classification with convolutional neural networks," in *Machine Learning for Signal Processing (MLSP), 2015 IEEE 25th International Workshop on*. IEEE, 2015, pp. 1–6.

[82] E. Cakir, T. Heittola, H. Huttunen, and T. Virtanen, "Polyphonic sound event detection using multi label deep neural networks," in *Neural Networks (IJCNN), 2015 International Joint Conference on*. IEEE, 2015, pp. 1–7.

[83] N. D. Lane, P. Georgiev, and L. Qendro, "Deepear: robust smartphone audio sensing in unconstrained acoustic environments using deep learning," in *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 2015, pp. 283–294.

[84] B. Wilkinson, C. Ellison, E. T. Nykaza, A. P. Boedihardjo, A. Netchaev, Z. Wang, S. L. Bunkley, T. Oates, and M. G. Blevins, "Deep learning for unsupervised separation of environmental noise sources," *The Journal of the Acoustical Society of America*, vol. 141, no. 5, pp. 3964–3964, 2017.

[85] L. Palen and P. Dourish, "Unpacking privacy for a networked world," in *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 2003, pp. 129–136.

[86] E. Kim, S. Helal, and D. Cook, "Human activity recognition and pattern discovery," *IEEE Pervasive Computing*, vol. 9, no. 1, 2010.

[87] E. Nazerfard, B. Das, L. B. Holder, and D. J. Cook, "Conditional random fields for activity recognition in smart environments," in *Proceedings of the 1st ACM International Health Informatics Symposium*. ACM, 2010, pp. 282–286.

[88] L. Stankovic, V. Stankovic, J. Liao, and C. Wilson, "Measuring the energy intensity of domestic activities from smart meter data," *Applied Energy*, vol. 183, pp. 1565–1580, 2016.

[89] A. Lavin and D. Klabjan, "Clustering time-series energy data from smart meters," *Energy efficiency*, vol. 8, no. 4, pp. 681–689, 2015.

[90] P. Cottone, S. Gaglio, G. L. Re, and M. Ortolani, "User activity recognition for energy saving in smart homes," *Pervasive and Mobile Computing*, vol. 16, pp. 156–170, 2015.

[91] S. Xu, E. Barbour, and M. C. González, "Household segmentation by load shape and daily consumption," in *Proceedings of ACM SigKDD 2017 conference, Halifax, Nova Scotia, Canada*, 2017.

[92] K. M. Rao, D. Ravichandran, and K. Mahesh, "Non-intrusive load monitoring and analytics for device prediction," in *Proceedings of the International MultiConference of Engineers and Computer Scientists*, vol. 1, 2016.

[93] A. Deshmukh and D. Lohan, "Cs446 project: Electric load identification using machine learning," University of Illinois Urbana-Champaign, Tech. Rep., 2015.

[94] C. Belley, S. Gaboury, B. Bouchard, and A. Bouzouane, "An efficient and inexpensive method for activity recognition within a smart home based on

load signatures of appliances," *Pervasive and Mobile Computing*, vol. 12, pp. 58 – 78, 2014.

[95] L. Chen, C. D. Nugent, and H. Wang, "A knowledge-driven approach to activity recognition in smart homes," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 6, pp. 961–974, 2012.

[96] J. Kelly and W. Knottenbelt, "Neural nilm: Deep neural networks applied to energy disaggregation," in *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*. ACM, 2015, pp. 55–64.

[97] C. Zhang, M. Zhong, Z. Wang, N. Goddard, and C. Sutton, "Sequence-to-point learning with neural networks for non-intrusive load monitoring," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[98] O. Krystalakos, C. Nalmpantis, and D. Vrakas, "Sliding window approach for online energy disaggregation using artificial neural networks," in *Proceedings of the 10th Hellenic Conference on Artificial Intelligence*. ACM, 2018, p. 7.

[99] F. G. Mármol, C. Sorge, O. Ugus, and G. M. Pérez, "Do not snoop my habits: preserving privacy in the smart grid," *IEEE Communications Magazine*, vol. 50, no. 5, pp. 166–172, 2012.

[100] A. Ukil, S. Bandyopadhyay, and A. Pal, "Privacy for iot: Involuntary privacy enablement for smart energy systems," in *2015 IEEE International Conference on Communications (ICC)*. IEEE, 2015, pp. 536–541.

[101] W. Asif, M. Rajarajan, and M. Lestas, "Increasing user controllability on device specific privacy in the internet of things," *Computer Communications*, vol. 116, pp. 200–211, 2018.

[102] C. A. Ronao and S.-B. Cho, "Human activity recognition with smartphone sensors using deep learning neural networks," *Expert Systems with Applications*, vol. 59, pp. 235–244, 2016.

[103] M. M. Hassan, M. Z. Uddin, A. Mohamed, and A. Almogren, "A robust human activity recognition system using smartphone sensors and deep learning," *Future Generation Computer Systems*, vol. 81, pp. 307–313, 2018.

[104] S. Blackman, C. Matlo, C. Bobrovitskiy, A. Waldoch, M. L. Fang, P. Jackson, A. Mihailidis, L. Nygård, A. Astell, and A. Sixsmith, "Ambient Assisted

Living Technologies for Aging Well: A Scoping Review," *Journal of Intelligent Systems*, vol. 25, no. 1, pp. 55–69, 2016.

[105] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, may 2015.

[106] Q. Zhu, Z. Chen, and Y. C. Soh, "A Novel Semi-supervised Deep Learning Method for Human Activity Recognition," *IEEE Transactions on Industrial Informatics*, vol. PP, no. c, p. 1, 2018.

[107] J. Morales and D. Akopian, "Physical activity recognition by smartphones, a survey," *Biocybernetics and Biomedical Engineering*, vol. 37, no. 3, pp. 388–400, 2017.

[108] M. Janidarmian, A. R. Fekr, K. Radecka, and Z. Zilic, "A comprehensive analysis on wearable acceleration sensors in human activity recognition," *Sensors*, vol. 17, no. 3, 2017.

[109] M. Espinilla, J. Medina, A. Salguero, N. Irvine, M. Donnelly, I. Cleland, and C. Nugent, "Human Activity Recognition from the Acceleration Data of a Wearable Device. Which Features Are More Relevant by Activities?" *Proceedings*, vol. 2, no. 19, p. 1242, 2018.

[110] F. Li, K. Shirahama, M. A. Nisar, L. Köping, and M. Grzegorzek, "Comparison of feature learning methods for human activity recognition using wearable sensors," *Sensors*, vol. 18, no. 2, pp. 1–22, 2018.

[111] A. Bulling, U. Blanke, and B. Schiele, "A tutorial on human activity recognition using body-worn inertial sensors," *ACM Computing Surveys (CSUR)*, vol. 1, no. June, pp. 1–33, 2014.

[112] S. Dernbach, B. Das, N. C. Krishnan, B. L. Thomas, and D. J. Cook, "Simple and Complex Activity Recognition through Smart Phones," *2012 Eighth International Conference on Intelligent Environments*, no. July 2017, pp. 214–221, 2012.

[113] E. Zdravevski, P. Lameski, V. Trajkovik, A. Kulakov, I. Chorbev, R. Goleva, N. Pombo, and N. Garcia, "Improving activity recognition accuracy in ambient-assisted living systems by automated feature engineering," *Ieee Access*, vol. 5, pp. 5262–5280, 2017.

[114] B. Hoh, M. Gruteser, H. Xiong, and A. Alrabady, "Preserving privacy in gps traces via uncertainty-aware path cloaking," in *Proceedings of the 14th ACM conference on Computer and communications security*, 2007, pp. 161–171.

[115] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "A public domain dataset for human activity recognition using smartphones." in *ESANN*, 2013.

[116] M. Perttunen, M. Van Kleek, O. Lassila, and J. Riekki, "Auditory context recognition using SVMs," in *Mobile Ubiquitous Computing, Systems, Services and Technologies, 2008. UBICOMM'08.* IEEE, 2008, pp. 102–108.

[117] X. Valero and F. Alias, "Gammatone cepstral coefficients: Biologically inspired features for non-speech audio classification," *IEEE Transactions on Multimedia*, vol. 14, no. 6, pp. 1684–1689, 2012.

[118] O. Abdel-Hamid, A.-r. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE/ACM Transactions on audio, speech, and language processing*, vol. 22, no. 10, pp. 1533–1545, 2014.

[119] E. M. Grais, H. Wierstorf, D. Ward, and M. D. Plumbley, "Multi-resolution fully convolutional neural networks for monaural audio source separation," in *International Conference on Latent Variable Analysis and Signal Separation.* Springer, 2018, pp. 340–350.

[120] V. Morfi and D. Stowell, "Deep learning for audio event detection and tagging on low-resource datasets," *Applied Sciences*, vol. 8, no. 8, p. 1397, 2018.

[121] H. Wang, D. Chong, D. Huang, and Y. Zou, "What affects the performance of convolutional neural networks for audio event classification," in *2019 8th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW).* IEEE, 2019, pp. 140–146.

[122] A. Vafeiadis, K. Votis, D. Giakoumis, D. Tzovaras, L. Chen, and R. Hamzaoui, "Audio content analysis for unobtrusive event detection in smart homes," *Engineering Applications of Artificial Intelligence*, vol. 89, p. 103226, 2020.

[123] Robinhood76, "Kitchen common sounds," 2008. [Online]. Available: https://www.freesound.org/people/Robinhood76/packs/3870

[124] J. Salamon and J. P. Bello, "Deep convolutional neural networks and data augmentation for environmental sound classification," *IEEE Signal Processing Letters*, vol. 24, no. 3, pp. 279–283, 2017.

[125] B. McFee, E. J. Humphrey, and J. P. Bello, "A software framework for musical data augmentation." in *ISMIR*, 2015, pp. 248–254.

[126] B. Milner, J. Darch, I. Almajai, and S. Vaseghi, "Comparing noise compensation methods for robust prediction of acoustic speech features from mfcc vectors in noise," in *2008 16th European Signal Processing Conference*, Aug. 2008, pp. 1–5.

[127] S. Furui, "Speaker-independent isolated word recognition based on emphasized spectral dynamics," in *ICASSP'86. IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 11. IEEE, 1986, pp. 1991–1994.

[128] L. Rabiner, *Fundamentals of speech recognition*. PTR Prentice Hall, 1993.

[129] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "librosa: Audio and music signal analysis in python," in *Proceedings of the 14th python in science conference*, 2015, pp. 18–25.

[130] S. Bilgin, O. Polat, and O. H. Colak, "The impact of daubechies wavelet performances on ventricular tachyarrhythmia patients for determination of dominant frequency bands in HRV," in *2009 14th National Biomedical Engineering Meeting*, May 2009, pp. 1–4.

[131] A. Jain and D. Zongker, "Feature selection: evaluation, application, and small sample performance," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 2, pp. 153–158, Feb. 1997.

[132] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[133] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International Conference on Machine Learning*, 2015, pp. 448–456.

[134] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (eLUs)," *arXiv preprint arXiv:1511.07289*, 2015.

[135] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting." *Journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[136] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proceedings of the 3rd International Conference for Learning Representations (ICLR-15)*, 2014.

[137] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers, "Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1285–1298, 2016.

[138] "scikit-learn. machine learning in python," 2019. [Online]. Available: https://scikit-learn.org/stable

[139] T. M. Oshiro, P. S. Perez, and J. A. Baranauskas, "How many trees in a random forest?" in *International workshop on machine learning and data mining in pattern recognition*. Springer, 2012, pp. 154–168.

[140] F. Chollet *et al.*, "Keras," https://github.com/fchollet/keras, 2015.

[141] A. Sholokhov, M. Sahidullah, and T. Kinnunen, "Semi-supervised speech activity detection with an application to automatic speaker verification," *Computer Speech & Language*, vol. 47, pp. 132–156, 2018.

[142] W. Xiong, L. Wu, F. Alleva, J. Droppo, X. Huang, and A. Stolcke, "The microsoft 2017 conversational speech recognition system," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5934–5938.

[143] D. Garcia-Romero, D. Snyder, G. Sell, D. Povey, and A. McCree, "Speaker diarization using deep neural network embeddings," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 4930–4934.

[144] T. Ng, B. Zhang, L. Nguyen, S. Matsoukas, X. Zhou, N. Mesgarani, K. Veselỳ, and P. Matějka, "Developing a speech activity detection system for the darpa rats program," in *Thirteenth Annual Conference of the International Speech Communication Association*, 2012.

[145] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The third 'chime'speech separation and recognition challenge: Dataset, task and baselines," in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE, 2015, pp. 504–511.

[146] H. Dubey, A. Sangwan, and J. H. Hansen, "Robust feature clustering for unsupervised speech activity detection," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 2726–2730.

[147] J. H. Hansen, A. Joglekar, M. Chandra Shekhar, V. Kothapally, C. Yu, L. Kaushik, and A. Sangwan, "The 2019 inaugural fearless steps challenge: A giant leap for naturalistic audio," *Proc. Interspeech 2019*, 2019.

[148] Y. Ephraim and D. Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator," *IEEE Transactions on Acoustics, Speech, and Signal processing*, vol. 32, no. 6, pp. 1109–1121, 1984.

[149] D. Malah, R. V. Cox, and A. J. Accardi, "Tracking speech-presence uncertainty to improve speech enhancement in non-stationary noise environments," in *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (ICASSP)*, vol. 2. IEEE, 1999, pp. 789–792.

[150] T. Gerkmann, C. Breithaupt, and R. Martin, "Improved a posteriori speech presence probability estimation based on a likelihood ratio with fixed priors," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 5, pp. 910–919, 2008.

[151] R. Martin and I. Cohen, "Single-channel speech presence probability estimation and noise tracking," in *Audio Source Separation and Speech Enhancement*. Wiley, 2018, ch. 6, pp. 87–106.

[152] J. Ramırez, J. C. Segura, C. Benıtez, A. De La Torre, and A. Rubio, "Efficient voice activity detection algorithms using long-term speech information," *Speech Communication*, vol. 42, no. 3-4, pp. 271–287, 2004.

[153] P. K. Ghosh, A. Tsiartas, and S. Narayanan, "Robust voice activity detection using long-term signal variability," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 3, pp. 600–613, 2011.

[154] G. Evangelopoulos and P. Maragos, "Speech event detection using multiband modulation energy," in *Ninth European Conference on Speech Communication and Technology*, 2005.

[155] H. Ghaemmaghami, B. Baker, R. Vogt, and S. Sridharan, "Noise robust voice activity detection using features extracted from the time-domain autocorrelation function," in *Eleventh Annual Conference of the International Speech Communication Association*, 2010.

[156] A. Saito, Y. Nankaku, A. Lee, and K. Tokuda, "Voice activity detection based on conditional random fields using multiple features," in *Eleventh Annual Conference of the International Speech Communication Association*, 2010.

[157] M. Graciarena, A. Alwan, D. Ellis, H. Franco, L. Ferrer, J. H. Hansen, A. Janin, B. S. Lee, Y. Lei, V. Mitra *et al.*, "All for one: feature combination for highly channel-degraded speech activity detection." in *Interspeech*, 2013, pp. 709–713.

[158] X. Wu, M. Zhu, R. Wu, and X. Zhu, "A self-adapting gmm based voice activity detection," in *2018 IEEE 23rd International Conference on Digital Signal Processing (DSP)*. IEEE, 2018, pp. 1–5.

[159] B. Liu, Z. Wang, S. Guo, H. Yu, Y. Gong, J. Yang, and L. Shi, "An energy-efficient voice activity detector using deep neural networks and approximate computing," *Microelectronics Journal*, 2019.

[160] F. Vesperini, P. Vecchiotti, E. Principi, S. Squartini, and F. Piazza, "Deep neural networks for multi-room voice activity detection: Advancements and comparative evaluation," in *2016 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2016, pp. 3391–3398.

[161] T. Hughes and K. Mierle, "Recurrent neural networks for voice activity detection," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2013, pp. 7378–7382.

[162] G. Gelly and J.-L. Gauvain, "Optimization of RNN-based speech activity detection," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 3, pp. 646–656, 2018.

[163] A. Vafeiadis, E. Fanioudakis, I. Potamitis, K. Votis, D. Giakoumis, D. Tzovaras, L. Chen, and R. Hamzaoui, "Two-Dimensional Convolutional

Recurrent Neural Networks for Speech Activity Detection," in *Proc. Interspeech 2019*, 2019, pp. 2045–2049. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2019-1354

[164] K. K. Paliwal and L. Alsteris, "Usefulness of phase spectrum in human speech perception," in *Eighth European Conference on Speech Communication and Technology*, 2003.

[165] M. Abadi *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[166] C. Bartz, T. Herold, H. Yang, and C. Meinel, "Language identification using deep convolutional recurrent neural networks," in *International Conference on Neural Information Processing*. Springer, 2017, pp. 880–889.

[167] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 2010, pp. 807–814.

[168] J. H. Hansen, A. Sangwan, A. Joglekar, A. E. Bulut, L. Kaushik, and C. Yu, "Fearless steps: Apollo-11 corpus advancements for speech technologies from earth to the moon," *Proc. Interspeech 2018*, pp. 2758–2762, 2018.

[169] "Google WebRTC," 2016. [Online]. Available: https://webrtc.org/

[170] B. A. Hanson and T. H. Applebaum, "Robust speaker-independent word recognition using static, dynamic and acceleration features: Experiments with lombard and noisy speech," in *International Conference on Acoustics, Speech, and Signal Processing*. IEEE, 1990, pp. 857–860.

[171] H. Zhao, S. Zarar, I. Tashev, and C.-H. Lee, "Convolutional-recurrent neural networks for speech enhancement," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 2401–2405.

[172] G. Papamakarios, D. Giakoumis, M. Vasileiadis, A. Drosou, and D. Tzovaras, "16 human computer confluence in the smart home paradigm: Detecting human states and behaviours for 24/7 support of mild-cognitive impairments," *Human Computer Confluence Transforming Human Experience Through Symbiotic Technologies*, pp. 275–293, 2016.

[173] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the fifth annual workshop on Computational learning theory*. ACM, 1992, pp. 144–152.

[174] S. Jun Lee and K. Siau, "A review of data mining techniques," *Industrial Management & Data Systems*, vol. 101, no. 1, pp. 41–46, 2001.

[175] I. Rish, "An empirical study of the naive bayes classifier," in *IJCAI 2001 workshop on empirical methods in artificial intelligence*, vol. 3, no. 22. IBM, 2001, pp. 41–46.

[176] H.-F. Yu, F.-L. Huang, and C.-J. Lin, "Dual coordinate descent methods for logistic regression and maximum entropy models," *Machine Learning*, vol. 85, no. 1, pp. 41–75, 2011.

[177] D. E. Rumelhart, G. E. Hinton, R. J. Williams *et al.*, "Learning representations by back-propagating errors," *Cognitive modeling*, vol. 5, no. 3, p. 1, 1988.

[178] A. Liaw, M. Wiener *et al.*, "Classification and regression by randomforest," *R news*, vol. 2, no. 3, pp. 18–22, 2002.

[179] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of statistics*, pp. 1189–1232, 2001.

[180] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[181] A. Mesaros, T. Heittola, A. Diment, B. Elizalde, A. Shah, E. Vincent, B. Raj, and T. Virtanen, "Dcase 2017 challenge setup: Tasks, datasets and baseline system," in *DCASE 2017-Workshop on Detection and Classification of Acoustic Scenes and Events*, 2017.

[182] D. Poirier, F. Fessant, and I. Tellier, "Reducing the cold-start problem in content recommendation through opinion classification," in *2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, vol. 1. IEEE, 2010, pp. 204–207.

[183] S. Hershey, S. Chaudhuri, D. P. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold *et al.*, "Cnn architectures for large-scale audio classification," in *2017 ieee international conference on acoustics, speech and signal processing (icassp)*. IEEE, 2017, pp. 131–135.

[184] F. J. Ordóñez and D. Roggen, "Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition," *Sensors*, vol. 16, no. 1, 2016.

[185] S.-J. Huang, W. Gao, and Z.-H. Zhou, "Fast multi-instance multi-label learning," *IEEE transactions on pattern analysis and machine intelligence*, 2018.

[186] K. Yordanova, A. Paiement, M. Schröder, E. Tonkin, P. Woznowski, C. M. Olsson, J. Rafferty, and T. Sztyler, "Challenges in annotation of user data for ubiquitous systems: Results from the 1st arduous workshop," *arXiv preprint arXiv:1803.05843*, 2018.

[187] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, pp. 160–167.

[188] F. Sebastiani, "Machine learning in automated text categorization," *ACM computing surveys (CSUR)*, vol. 34, no. 1, pp. 1–47, 2002.

[189] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *Journal of Machine Learning Research*, vol. 12, no. Aug, pp. 2493–2537, 2011.

[190] W. Huang, "Character-level convolutional network for text classification applied to chinese corpus," Ph.D. dissertation, University College London, 2016.

[191] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[192] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder–decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1724–1734.

[193] M. Seo, S. Min, A. Farhadi, and H. Hajishirzi, "Neural speed reading via skim-RNN," in *International Conference on Learning Representations*, 2018.

[194] T. Trinh, A. Dai, T. Luong, and Q. Le, "Learning longer-term dependencies in RNNs with auxiliary losses," in *Proceedings of the 35th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, J. Dy and A. Krause, Eds., vol. 80. Stockholmsmässan, Stockholm Sweden: PMLR, 10–15 Jul 2018, pp. 4965–4974.

[195] A. W. Yu, H. Lee, and Q. Le, "Learning to skim text," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, vol. 1, 2017, pp. 1880–1890.

[196] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *International Conference on Learning Representations*, 2015.

[197] T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2015, pp. 1412–1421.

[198] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 6000–6010.

[199] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[200] ——, "Identity mappings in deep residual networks," in *European Conference on Computer Vision*. Springer, 2016, pp. 630–645.

[201] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1251–1258.

[202] Z. S. Harris, "Distributional structure," *Word*, vol. 10, no. 2-3, pp. 146–162, 1954.

[203] S. Koelsch, E. Kasper, D. Sammler, K. Schulze, T. Gunter, and A. D. Friederici, "Music, language and meaning: brain signatures of semantic processing," *Nature neuroscience*, vol. 7, no. 3, p. 302, 2004.

[204] P. G. Simos, L. F. Basile, and A. C. Papanicolaou, "Source localization of the n400 response in a sentence-reading paradigm using evoked magnetic fields and magnetic resonance imaging," *Brain research*, vol. 762, no. 1-2, pp. 29–39, 1997.

[205] Y. Kim, "Convolutional neural networks for sentence classification," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1746–1751.

[206] P. Blunsom, E. Grefenstette, and N. Kalchbrenner, "A convolutional neural network for modelling sentences," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, 2014.

[207] P. Wang, J. Xu, B. Xu, C. Liu, H. Zhang, F. Wang, and H. Hao, "Semantic clustering and convolutional neural network for short text categorization," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, vol. 2, 2015, pp. 352–357.

[208] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," in *Advances in neural information processing systems*, 2015, pp. 649–657.

[209] A. Conneau, H. Schwenk, L. Barrault, and Y. Lecun, "Very deep convolutional networks for text classification," in *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, vol. 1, 2017, pp. 1107–1116.

[210] C. N. Dos Santos and M. Gatti, "Deep convolutional neural networks for sentiment analysis of short texts." in *COLING*, 2014, pp. 69–78.

[211] R. Johnson and T. Zhang, "Effective use of word order for text categorization with convolutional neural networks," in *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2015, pp. 103–112.

[212] ——, "Semi-supervised convolutional neural networks for text categorization via region embedding," in *Advances in neural information processing systems*, 2015, pp. 919–927.

[213] S. Ruder, P. Ghaffari, and J. G. Breslin, "Character-level and multi-channel convolutional neural networks for large-scale authorship attribution," *arXiv preprint arXiv:1609.06686*, 2016.

[214] J. Bjerva, B. Plank, and J. Bos, "Semantic tagging with deep residual networks," in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, 2016, pp. 3531–3541.

[215] R. Lowe, I. V. Serban, M. Noseworthy, L. Charlin, and J. Pineau, "On the evaluation of dialogue systems with next utterance classification," *Proceedings of the 2016 SIGDIAL*, 2016.

[216] I. V. Serban, A. Sordoni, Y. Bengio, A. Courville, and J. Pineau, "Building end-to-end dialogue systems using generative hierarchical neural network models," in *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[217] I. V. Serban, A. Sordoni, R. Lowe, L. Charlin, J. Pineau, A. Courville, and Y. Bengio, "A hierarchical latent variable encoder-decoder model for generating dialogues," in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[218] J. Li, W. Monroe, T. Shi, S. Jean, A. Ritter, and D. Jurafsky, "Adversarial learning for neural dialogue generation," *arXiv preprint arXiv:1701.06547*, 2017.

[219] S. Sukhbaatar, J. Weston, R. Fergus *et al.*, "End-to-end memory networks," in *Advances in neural information processing systems*, 2015, pp. 2440–2448.

[220] J. Weston, S. Chopra, and A. Bordes, "Memory networks," *arXiv preprint arXiv:1410.3916*, 2014.

[221] A. Bordes and J. Weston, "Learning end-to-end goal-oriented dialog," in *International Conference on Learning Representations*, 2017.

[222] S. Cèbe and R. Goigoux, *Apprendre à lire à l'école: Tout ce qu'il faut savoir pour accompagner l'enfant.* Retz, 2011.

[223] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning." in *AAAI*, vol. 4, 2017, p. 12.

[224] A. Copestake, "Augmented and alternative nlp techniques for augmentative and alternative communication," *Natural Language Processing for Communication Aids*, 1997.

[225] Í. de Pontes Oliveira, J. L. P. Medeiros, V. F. de Sousa, A. G. T. Júnior, E. T. Pereira, and H. M. Gomes, "A data augmentation methodology to improve age estimation using convolutional neural networks," in *2016 29th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*. IEEE, 2016, pp. 88–95.

[226] J. McAuley and J. Leskovec, "Hidden factors and hidden topics: understanding rating dimensions with review text," in *Proceedings of the 7th ACM conference on Recommender systems*. ACM, 2013, pp. 165–172.